文章编号:1674-2974(2024)12-0153-12 DOI:10.16339/j.cnki.hdxbzkb.2024292

面向零日攻击检测的APT攻击活动辨识研究

成翔[1,2†],匡苗苗[1],严莉萍[3],张佳乐[1],杨宏宇[4]
(1. 扬州大学 信息工程学院,江苏 扬州 225127;
2. 中国民航大学 民航飞联网重点实验室,天津 300300;
3. 民航机场成都电子工程设计有限责任公司,四川 成都 610042;
4. 中国民航大学 安全科学与工程学院,天津 300300)

摘 要:传统的攻击检测方法很难辨识出利用零日漏洞发起的高级持续性威胁(advanced persistent threat, APT)攻击活动.为此提出一种面向零日攻击检测的APT攻击活动辨识方法(APTIZDM),该方法由三个主要部分组成.第一部分态势觉察本体构建(CSPOC)方法进行物联网(IoT)系统中关键活动属性及特征的形式化描述.第二部分恶意C&C(command and control)DNS响应活动挖掘(MCCDRM)方法用于辨识APT攻击情境中的恶意C&C通信活动,并可有效控制活动辨识过程的范围与起始时间,从而减小计算开销.第三部分APT攻击情境中零日攻击活动辨识(ZDAARA)方法,其基于贝叶斯网络和安全风险传播理论,对系统调用信息进行关联分析,计算出各系统调用实例的恶意概率,可有效辨识出被入侵检测系统漏报的零日攻击活动.仿真实验结果表明,作为APTIZDM的核心内容,MCCDRM方法和ZDAARA方法都实现了较高的准确率和较低的误报率,协同完成了对APT攻击活动有效辨识.

关键词:零日攻击;边缘计算;贝叶斯网络;C&C
中图分类号:TP393 **文献标志码**:A

An APT Attack Activity Identification Research for Zero Day Attack Detection

CHENG Xiang[1,2†], KUANG Miaomiao[1], YAN Liping[3], ZHANG Jiale[1], YANG Hongyu[4]
(1. College of Information Engineering, Yangzhou University, Yangzhou 225127, China;
2. Key Laboratory of Flying Internet, Civil Aviation University of China, Tianjin 300300, China;
3. Civil Aviation Airport Chengdu Electronic Engineering Design Co., Ltd., Chengdu 610042, China;
4. School of Safety Science and Engineering, Civil Aviation University of China, Tianjin 300300, China)

Abstract: The traditional attack detection methods struggle to identify advanced persistent threat (APT)

* **收稿日期**:2023-11-16
基金项目:江苏省基础研究计划青年基金项目(BK20230558), Youth Foundation of Jiangsu Basic Research Program(BK20230558);新疆维吾尔自治区自然科学基金项目(2024D01A40), Natural Science Foundation of Xinjiang Uygur Autonomous Region(2024D01A40);民航机场工程技术研究中心开放课题(ERCAOTP20230301), Open Project of Civil Aviation Airport Engineering Technology Research Center(ERCAOTP20230301);中国民航大学民航飞联网重点实验室开放基金(MHFLW202304), Open Fund for the Key Laboratory of Flying Internet at Civil Aviation University of China (MHFLW202304)
作者简介:成翔(1988—),男,新疆乌鲁木齐人,扬州大学实验师,博士
† 通信联系人,E-mail: huozhai9527@126.com

attacks launched using zero-day vulnerabilities. To address this issue, this paper proposes an APT attack activity identification for zero-day attack method (APTIZDM), which consists of three key components. The first component is the cyber situation perception ontology construction (CSPOC) method, which provides a formal description of critical activity attributes and features in IoT systems. The second component is the malicious command & control (C&C) DNS response activity mining (MCCDRM) method, which identifies malicious C&C communication activities in APT attack scenarios while effectively controlling the scope and starting time of the identification process, thereby reducing computational overhead. The third component is the zero-day attack activity recognition method in APT attack (ZDAARA) scenarios, which utilizes Bayesian networks and security risk propagation theory to perform correlation analysis on system call information. It calculates the malicious probability of each system call instance and effectively identifies zero-day attack activities missed by intrusion detection systems. Simulation experiment results demonstrate that MCCDRM and ZDAARA, as the core components of the APTIZDM, achieve high accuracy and low false positive rates, effectively collaborating to identify APT attack activities.

Key words: zero-day attack; edge computing; Bayesian networks; command and control

随着网络基础设施性能提升,信息技术广泛应用,传统活动的时间和空间限制被打破.然而,在信息技术及应用快速发展的同时,各类网络信息系统也面临着大量的信息安全风险和隐患[1-5].其中,高级持续性威胁(advanced persistent threat, APT)的定义是结合高级、持续性和威胁3个词而确定的[6].根据美国国家标准与技术研究院[7]的定义,APT有以下特点:1)APT攻击者会长期追踪其攻击目标;2)APT攻击者通过不断适应目标系统的安全防御策略来驻留于目标系统中;3)APT攻击者会采用多种手段保护好在实现最终攻击目标过程所需的系统交互权限.为了有效抵御APT攻击,提出并设计出一种可行的APT攻击活动辨识方法成为对信息系统进行安全防御的迫切需求.

然而,传统的网络攻击活动辨识方法在应对APT攻击时会面临着诸多挑战:1)APT的一些攻击活动大多是利用零日漏洞完成对目标系统的首次突破,难以被传统的入侵检测系统拦截.2)大多数网络攻击辨识方法是离线的,无法满足对APT攻击的及时响应与防御需求.3)绝大多数分散部署的终端设备缺乏足够计算资源,因此将恶意活动语义挖掘任务分配给各终端设备是不可行的.

为了应对并解决上述挑战,本文提出了一种新颖的面向零日攻击检测的APT攻击活动辨识方法(APT attack activity identification for zero-day attack method, APTIZDM).鉴于APT攻击情境实质上是一条可检测的恶意活动与零日攻击活动组成的攻击链,零日攻击活动和这些被检测到的恶意活动之间存在确定的链式关系,将APT攻击情境中可检测到的恶意活动连接成串,通过有效的方法挖掘出隐藏在此攻击情境中的零日攻击活动片段,是一种简单而直接的辨识策略.

因此,本文提出的APTIZDM主要包括了2个子方法:1)提出了一种恶意C&C DNS响应活动挖掘(MCCDRM)方法,用于辨识APT攻击情境中的恶意C&C通信活动,并可用于触发零日攻击辨识过程,大幅度降低了整个信息系统在应用零日攻击辨识方法时的计算开销.2)提出了一种APT攻击情境中零日攻击活动辨识(ZDAARA)方法,用于挖掘并检测出那些因利用零日漏洞而无法被入侵检测系统(intrusion detection system, IDS)发觉的恶意活动.本文基于边缘计算架构,将APTIZDM应用于典型物联网(IoT)系统.边缘服务器在该架构中的作用不仅是分担终端设备在恶意攻击活动语义挖掘方面的计算开销,还能减轻终端设备与云端之间的通信开销.文章主要贡献和创新点如下.

1)本文提出的APTIZDM可对APT攻击情境中的恶意C&C通信活动和零日攻击活动进行准确、及

时和有效的辨识,并且该方法可以很好地应用于基于边缘计算架构的物联网系统.

2)MCCDRM作为APTIZDM的核心子方法,它基于域名与恶意IP地址间的映射关系,可用于辨识APT攻击情境中极具代表性的恶意C&C通信活动,并可对ZDAARA方法所涉及的关联分析进行触发控制,有效降低边缘服务器的计算开销.

3)ZDAARA作为APTIZDM的另一个核心子方法,它首先通过建立系统调用实例图捕获物联网系统中的关键活动,呈现出安全风险在系统调用实例间可能的传播路径.然后,通过将告警信息融入基于系统实例图建立的网络活动贝叶斯图,推演筛选出恶意概率高于恶意阈值的系统调用实例,从而辨识出隐藏在其中的零日攻击活动.

1　方法模型与架构

1.1　面临的主要挑战

如果将传统的攻击检测方法应用于APT攻击活动辨识,将面临三大挑战:零日攻击辨识、实时监测、硬件资源受限.

1.2　方法的原理及假设

为了应对上述挑战,本文根据以下三个方面设计原理进行研究.

1.2.1　对零日攻击进行有效辨识

由于现有的网络安全态势觉察方法很难对零日攻击进行有效辨识,本文提出了一种APT攻击情境中零日攻击活动辨识(ZDAARA)方法,用于对那些无法通过入侵检测系统进行有效辨识的攻击活动进行监测与辨识.

1.2.2　恶意C&C DNS挖掘

实时的APT恶意活动关联分析操作难以实现,这已成为对物联网系统中恶意活动进行实时监测的重要阻碍,本文提出了一种恶意C&C DNS响应活动挖掘(MCCDRM)方法,辨识APT攻击情境中的恶意C&C通信活动以及为零日攻击活动辨识(ZDAARA)提供触发机制,缓解了时刻进行恶意活动挖掘所导致的计算资源过度消耗.

1.2.3　基于边缘计算的架构

受到新兴的边缘计算架构的启发,本文进一步呈现了一种基于边缘计算的系统架构,用于将APTIZDM应用于典型的物联网系统中.此架构中的边缘服务器不但可以分担物联网终端设备的计算开销,还可以削减物联网终端设备与云服务器间的通信开销.

基于以上设计原理,本文作了如下假设:

1)可检测的恶意活动和零日攻击活动之间存在一定的关系.鉴于APT攻击情境实质上是一条可检测的恶意活动与零日攻击活动组成的攻击链.因此,两者之间存在某种链式关系.

2)系统调用信息具有攻击中立性.由于系统调用信息是一种用户与操作系统交互时不可避免的交互方式,所以系统调用信息能捕获到物联网系统中所有活动(包括良性与恶性活动).

3)攻击者不可避免与C&C服务器发起通信.当APT攻击者突破防线进入目标系统后,将很快试图在系统中建立后门,并且不可避免地需要通过C&C服务器控制系统中的相关实体,从而达到提高访问权限,为后续攻击活动做准备的目的.

4)利用恶意C&C通信触发零日攻击活动辨识操作.在整个APT攻击情境中,恶意C&C通信之前的攻击阶段对目标系统影响较小,而恶意C&C通信之后的攻击阶段对系统构成巨大威胁.在辨识出恶意C&C通信后,如果能完成对恶意系统调用实例的辨识,则可以实现系统安全与计算开销之间的平衡.

1.3　整体架构

基于上述原理与假设,本文提出了一种新颖的基于零日攻击活动辨识的物联网系统中APT攻击活动辨识方法(APTIZDM).APTIZDM的整体架构和各模块实现的主要功能如图1所示.与此同时,本文还提出了一种将此APT攻击活动辨识方法应用于基于边缘计算的物联网系统的架构,如图2所示.得益于边缘计算架构的显著特性,物联网终端设备对于恶意攻击活动语义挖掘的计算开销可以被与其连接的边缘服务器所分担.

本文所提出的APTIZDM首先需要定义一组表示原语,用于描述物联网系统中存在的关键网络活动.其中提出的态势觉察本体构建(CSPOC)方法,可

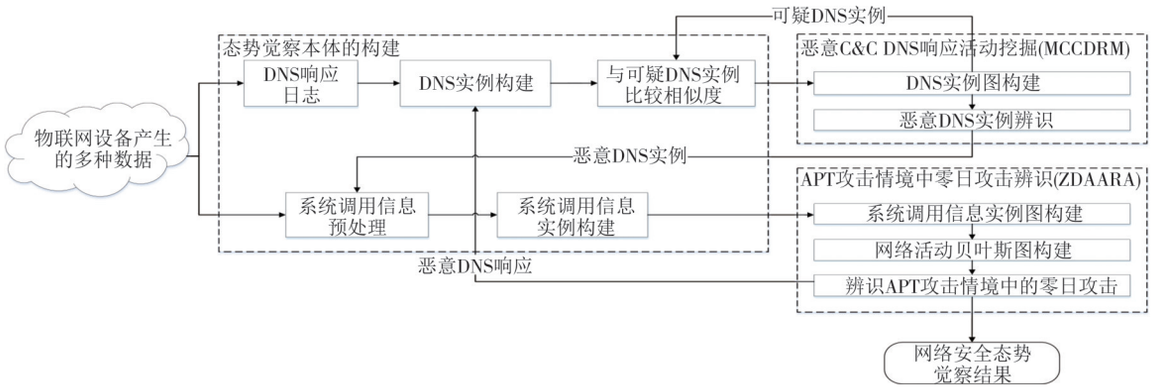


图 1 APTIZDM 的整体架构

Fig.1 The overall architecture of the APTIZDM

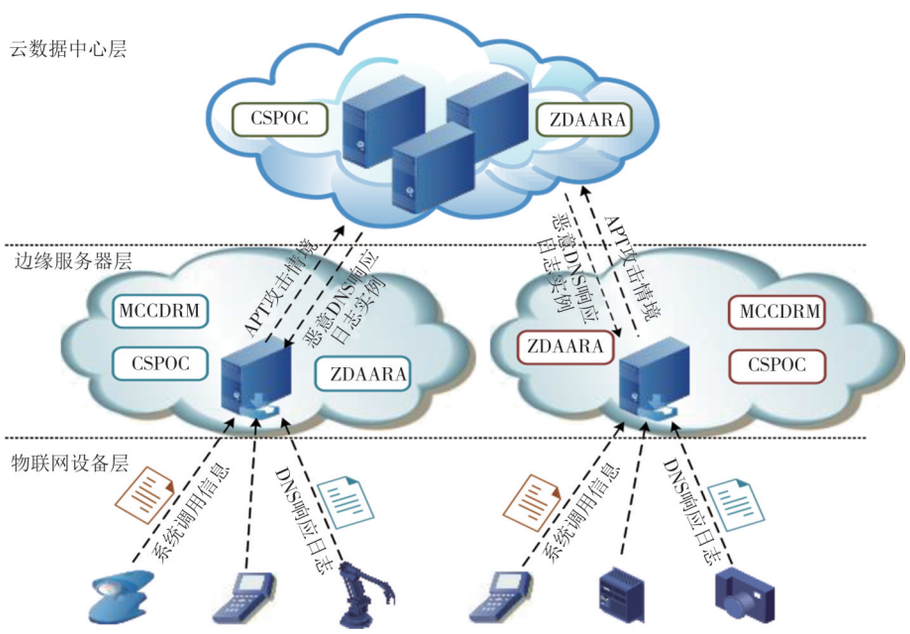


图 2 APTIZDM 应用于物联网系统的架构

Fig.2 The architecture of application of APTIZDM in the IoT system

以实现对网络安全态势觉察过程中涉及的属性和概念的规范化与形式化描述. 该方法主要用于将物联网终端设备中产生的系统调用信息和DNS响应日志转化为相应的实例. 与此同时, 在态势觉察本体构建过程中还将度量构建得到的DNS实例与可疑DNS实例之间的相似度, 用于后续的MCCDRM模块. 还应该在云数据中心部署CSPOC模块用于构建态势觉察实例, 这样可以有效地捕获潜在其中的零日攻击活动.

本文提出的MCCDRM方法, 可对隐藏在大量DNS响应日志中的恶意C&C DNS响应活动进行检测辨识. 首先, 该方法将度量新出现的DNS实例与入侵检测系统辨识出的恶意DNS实例之间的相似度. 如果两者之间存在相似性, 这一新出现的DNS实例会被添加到DNS实例图中, 并被视为可疑DNS实例. 同时, 这些可疑DNS实例还会被反馈至CSPOC模块, 用于更新实例相似度度量所需要的知识. 为了识别大量的可疑DNS实例中隐藏的恶意DNS实例, 构建得到的DNS实例图将被转化成一个域名无向图. 此图的构建规则为, 如果两个DNS实例指向同一个IP地址, 那么就在它们之间建立一条边. 紧接着, 将基于域名的不同计算每个DNS实例的恶意分值. 一旦恶意分值超过既定的阈值, 某个DNS实例就会被辨识为恶意DNS实例, 并且CSPOC模块也将开始对此DNS实例所属的物联网终端设备进行系统调用信息的采集.

本文提出的ZDAARA方法可有效辨识隐藏在大量系统调用实例中的零日攻击活动. 利用该方法实现的模块应当部署于边缘服务器和云数据中心, 挖

掘检测到的恶意系统调用实例与利用零日漏洞的系统调用实例之间的链接效应.通过借助贝叶斯网络和信息系统安全风险传播理论,ZDAARA方法具备了对APT攻击情境中零日攻击活动有效辨识的能力.

2 方法设计

在本节中,首先详细介绍了态势觉察感知本体的构建方法,用于对DNS日志和系统调用信息进行规范化处理.接着,介绍了如何挖掘APT攻击情境中的恶意C&C DNS响应活动,并实现对APT恶意活动关联分析操作的触发与控制.最后,阐述了如何实现对APT攻击情境中的零日攻击活动进行有效辨识.

2.1 态势觉察本体的构建

能够捕获物联网系统中关键活动的态势觉察本体构建(CSPOC)过程是攻击活动辨识过程中与粗糙的网络安全监测数据交互的第一个功能模块.该模块部署在边缘服务器和云数据中心中,承担对DNS响应日志和系统调用信息进行规范化的任务.为此,本文给出了态势觉察本体的统一定义,该本体由两类态势觉察实例组成:DNS实例与系统调用实例.在各类物联网终端设备中产生的DNS响应日志会由边缘服务器收集并存储在DNS日志数据集中.然后,边缘服务器中部署的CSPOC模块将负责对这些DNS日志进行处理并构建DNS实例.为了给后续恶意C&C DNS响应活动挖掘提供量化依据,CSPOC还对新产生的DNS实例与可疑DNS实例之间的相似度进行了量化度量.当新的恶意DNS实例被辨识出后,CSPOC模块便开始收集与其相关的系统调用信息,以捕获边缘服务器控制的物联网终端设备中产生的关键网络活动.这些系统调用信息也将根据既定的规则和表征原语转换成为系统调用实例.最终,构建得到的两种态势觉察实例可以作为后续MCCDRM模块和ZDAARA模块的输入数据.

2.1.1 态势觉察本体初始化

态势觉察本体初始化阶段确定从DNS响应日志和系统调用信息中提取哪些有效信息用于构建态势觉察实例.已有的研究工作已经定义出“本体”的概念[8]:当本体的概念应用于态势觉察领域时,本文提出了一种新颖的态势觉察本体,这种本体可以形式化地表示为:$\boldsymbol{O}=(\boldsymbol{C},\boldsymbol{P},\boldsymbol{D},\boldsymbol{R})$.其中,$\boldsymbol{C}$形式化地表示出了态势觉察过程涉及的两种概念:DNS响应日志(Dns)和系统调用信息(Sys);$\boldsymbol{P}$代表每种概念所包含的属性(参数);$\boldsymbol{D}$表示网络安全态势觉察信息;$\boldsymbol{R}$用于呈现本体中不同实体之间的相似度(主要用于度量DNS实例间的相似度).根据上述定义,采用表达式$\mathrm{In}(\boldsymbol{c}_i)_m$表示概念$\boldsymbol{c}_i$中包含的第$m$个实例,$P(\boldsymbol{c}_i)$表示概念$\boldsymbol{c}_i$中所需的参数.除此之外,本文还给出了某一实例参数值的形式化表示$P\left(\mathrm{In}(\boldsymbol{c}_i)_m\right)$,以及两个实例间相似度的形式化表示$\mathrm{REM}\left(P\left(\mathrm{In}(\boldsymbol{c}_i)_m\right),P\left(\mathrm{In}(\boldsymbol{c}_i)_n\right)\right)$.

在边缘服务器上收集和构建的DNS实例可以捕获物联网系统中与恶意C&C服务器的通信活动.作为这些实例的抽象表示,DNS概念应当给出确定的参数类型用于描述DNS实例的特征.在态势觉察本体构建过程中,本文确定了6个参数用于表示DNS响应活动的特征:$\mathrm{Tar_{host}}$、$\mathrm{Req_{type}}$、$\mathrm{Req_{IP}}$、$\mathrm{Req_{host}}$、$\mathrm{Req_{domain}}$和$\mathrm{Req_{time}}$(表1).本文设定了一个由6个参数组成的向量用于储存某一DNS实例中各个参数的值,表达式如下:

$$P\left(\mathrm{In(Dns)}_m\right)=\left(\mathrm{Tar_{host}},\mathrm{Req_{type}},\mathrm{Req_{IP}},\mathrm{Req_{host}},\mathrm{Req_{domain}},\mathrm{Req_{time}}\right)\tag{1}$$

表1 DNS实例参数

Tab.1 DNS instance parameters

序号	参数名	语义
1	$\mathrm{Tar_{host}}$	DNS响应活动出现的主机
2	$\mathrm{Req_{type}}$	DNS响应活动的类型
3	$\mathrm{Req_{IP}}$	由域名解析映射到的IP地址
4	$\mathrm{Req_{host}}$	发起DNS响应活动的主机名
5	$\mathrm{Req_{domain}}$	DNS响应活动的域名
6	$\mathrm{Req_{time}}$	DNS响应活动发起的时间

CSPOC模块通过查询syscall表来收集系统调用信息,该模块将获取的原始数据转换为系统调用实例,以捕获物联网系统中的关键网络活动,它侧重于提取系统调用信息中有价值的参数用于表征这些活动的关键特征.在这一过程中确定了7个参数用于表示恶意和良性系统调用信息的特征:$\mathrm{Sys_{type}}$、$\mathrm{Start_{time}}$、$\mathrm{End_{time}}$、$\mathrm{Pro_{ID}}$、$\mathrm{Pro_{name}}$、$\mathrm{File_{path}}$和$\mathrm{File_{inode}}$(表2).本文也设定了一个由7个参数组成的向量用于储存某一系统调用实例中各个参数的值,表达式如式(2)所示:

$$P\left(\mathrm{In(Sys)}_m\right)=\left(\mathrm{Sys_{type}},\mathrm{Start_{time}},\mathrm{End_{time}},\mathrm{Pro_{ID}},\mathrm{Pro_{name}},\mathrm{File_{path}},\mathrm{File_{inode}}\right)\tag{2}$$

表2 系统调用实例参数

Tab.2 System call instance parameters

序号	参数名	语义
1	Sys_{type}	系统调用类型
2	$Start_{time}$	系统调用开始时间
3	End_{time}	系统调用结束时间
4	Pro_{ID}	进程ID
5	Pro_{name}	进程名
6	$File_{path}$	系统调用作用的文件的路径
7	$File_{inode}$	系统调用活动的文件的索引节点

2.1.2 度量DNS实例间的相似度

基于以下两点事实，本文选择采用有向图来描述DNS实例之间的依赖关系和相似性：1)DNS实例是随着时间的推移而逐渐出现在物联网系统中的；2)不同实例之间可能存在潜在的依赖关系.因此，本文将实例间的相似度设置为这种有向图中边上的权重.通过结合图结构的有关特性，本文对SimRank算法[9]进行了改进，实现了对DNS实例间相似度的计算.由于不同的DNS实例包含相同的参数类型，这将非常便于使用SimRank算法度量每两个DNS实例间的相似度.为形式化地描述父实例与子实例之间的关系，本文设置了使用$\mathrm{Fa}\left(\mathrm{In}(\mathrm{Dns})_m\right)$表示实例$\mathrm{In}(\mathrm{Dns})_m$的父实例.

基于以上内容，本文提出了一种形式化的计算方法来度量两个DNS实例之间的相似度，这一DNS实例相似度度量方法主要包括参数相似度度量和父实例相似度度量.如式(3)所示，实例间相似度可由上述两部分相加而得到.

$$\begin{aligned}&\mathrm{REM}\left(\mathrm{In}(\mathrm{Dns})_m,\mathrm{In}(\mathrm{Dns})_n\right)=\\&\alpha\cdot\mathrm{REMF}\left(\mathrm{In}(\mathrm{Dns})_m,\mathrm{In}(\mathrm{Dns})_n\right)+\\&\mu\cdot\mathrm{REMP}\left(\mathrm{In}(\mathrm{Dns})_m,\mathrm{In}(\mathrm{Dns})_n\right)\\&\alpha=\\&\frac{\left|\mathrm{Fa}\left(\mathrm{In}(\mathrm{Dns})_m\right)\cup\mathrm{Fa}\left(\mathrm{In}(\mathrm{Dns})_n\right)\right|}{\left|\mathrm{Fa}\left(\mathrm{In}(\mathrm{Dns})_m\right)\cup\mathrm{Fa}\left(\mathrm{In}(\mathrm{Dns})_n\right)\right|+\left|P\left(\mathrm{In}(\mathrm{Dns})_m\right)\cup P\left(\mathrm{In}(\mathrm{Dns})_n\right)\right|}\\&\mu=\\&\frac{\left|P\left(\mathrm{In}(\mathrm{Dns})_m\right)\cup P\left(\mathrm{In}(\mathrm{Dns})_n\right)\right|}{\left|\mathrm{Fa}\left(\mathrm{In}(\mathrm{Dns})_m\right)\cup\mathrm{Fa}\left(\mathrm{In}(\mathrm{Dns})_n\right)\right|+\left|P\left(\mathrm{In}(\mathrm{Dns})_m\right)\cup P\left(\mathrm{In}(\mathrm{Dns})_n\right)\right|}\end{aligned}\tag{3}$$

式(3)中的调节参数α和μ可保证计算得到的相似度数值落入[0，1]区间.可通过递归计算得到两个特定DNS实例的父实例之间的相似程度，形式化描述如式(4)所示.若实例$\mathrm{In}(\mathrm{Dns})_m$或$\mathrm{In}(\mathrm{Dns})_n$在图中没有父实例节点，则父实例相似度为0.

$$\mathrm{REMF}\left(\mathrm{In}(\mathrm{Dns})_m,\mathrm{In}(\mathrm{Dns})_n\right)=\begin{cases}1, & \mathrm{In}(\mathrm{Dns})_m=\mathrm{In}(\mathrm{Dns})_n\\ \dfrac{\partial\sum\limits_{i=1}^{\left|\mathrm{Fa}\left(\mathrm{In}(\mathrm{Dns})_m\right)\right|}\sum\limits_{j=1}^{\left|\mathrm{Fa}\left(\mathrm{In}(\mathrm{Dns})_n\right)\right|}\mathrm{REMF}\left(\mathrm{Fa}_i\left(\mathrm{In}(\mathrm{Dns})_m\right),\mathrm{Fa}_j\left(\mathrm{In}(\mathrm{Dns})_n\right)\right)}{\left|\mathrm{Fa}\left(\mathrm{In}(\mathrm{Dns})_m\right)\right|\left|\mathrm{Fa}\left(\mathrm{In}(\mathrm{Dns})_n\right)\right|}, & \mathrm{In}(\mathrm{Dns})_m\neq\mathrm{In}(\mathrm{Dns})_n\end{cases}\tag{4}$$

$\partial\in[0,1]$，表示阻尼系数.参数相似度度量的数值结果可以通过递归比较每两个DNS实例各个参数的数值而计算得到.对于完全相同的两个实例，二者间的相似度可设为“1”，不同实例间的相似度的形式化描述如式(5)所示.式(6)进一步给出了如何对每两个参数进行比较.一旦在两个特定的DNS实例中无法得到相同的参数对，则将利用式(5)递归计算它们各自父实例之间的参数相似度.式(5)中的参数γ和τ也是用于将最终计算得到的相似度控制在一个相对合适的区间中.

$$\mathrm{REMP}\left(\mathrm{In}(\mathrm{Dns})_m,\mathrm{In}(\mathrm{Dns})_n\right)=\begin{cases}\dfrac{\tau\sum\limits_{i=1}^{\left|P\left(\mathrm{In}(\mathrm{Dns})\right)\right|}\delta\left(P_i\left(\mathrm{In}(\mathrm{Dns})_m\right),P_i\left(\mathrm{In}(\mathrm{Dns})_n\right)\right)}{\left|P\left(\mathrm{In}(\mathrm{Dns})\right)\right|}, & \exists\delta\left(P_i\left(\mathrm{In}(\mathrm{Dns})_m\right),P_i\left(\mathrm{In}(\mathrm{Dns})_n\right)\right)\neq0\\ \dfrac{\gamma\sum\limits_{i=1}^{\left|\mathrm{Fa}\left(\mathrm{In}(\mathrm{Dns})_m\right)\right|}\sum\limits_{j=1}^{\left|\mathrm{Fa}\left(\mathrm{In}(\mathrm{Dns})_n\right)\right|}\mathrm{REMP}\left(\mathrm{Fa}_i\left(\mathrm{In}(\mathrm{Dns})_m\right),\mathrm{Fa}_j\left(\mathrm{In}(\mathrm{Dns})_n\right)\right)}{\left|\mathrm{Fa}\left(\mathrm{In}(\mathrm{Dns})_m\right)\right|\left|\mathrm{Fa}\left(\mathrm{In}(\mathrm{Dns})_n\right)\right|}, & \forall\delta\left(P_i\left(\mathrm{In}(\mathrm{Dns})_m\right),P_i\left(\mathrm{In}(\mathrm{Dns})_n\right)\right)=0\end{cases}\tag{5}$$

$$\delta(a,b)=\begin{cases}1, a=b\\ 0, a\neq b\end{cases}\tag{6}$$

2.2 恶意C&C DNS响应活动挖掘

对某一APT攻击情境的C&C阶段生成的DNS实例，可以有效捕获其中的恶意通信活动.然而，可检测到的恶意DNS实例与可疑DNS实例之间确实存在潜在相似关系.通过比较新产生的DNS实例与检测到的恶意实例，从而逐步增加可疑DNS实例数量是一种极为可行的安全策略.具体而言，在某一恶意DNS挖掘时间窗口内，与检测到的恶意DNS实例存在相似性的实例可聚类成有向图的形式，并将它们视为可疑DNS实例.与此同时，存在于DNS实例图中的可疑DNS实例信息将被其发送回CSPOC模块用于更新后续相似度测量基准.本文将DNS实例图中所包含的域名和IP地址间的映射关系提取出来并构建出一种新的域名无向图.此图的构建规则为，如果两个DNS实例的Req_{domain}映射到相同的Req_{IP}，

则在两个实例间建立一条边.然后,基于不同Req_{domain}所映射到的相同Req_{IP}的数量进一步度量不同Req_{domain}间的关系,并计算得出每个Req_{domain}的恶意分值.一旦某个DNS实例的域名恶意分值超过阈值,该可疑DNS实例将被视为恶意DNS实例.

2.2.1　DNS实例图构建

本文提出的MCCDRM方法主要用于将物联网系统中出现的与恶意DNS实例具有相似性的DNS响应活动关联起来,从而挖掘出其中新产生的恶意DNS响应活动.当DNS实例图构建模块接收到一个已被辨识出的恶意DNS实例,这一实例将被作为一个新的根节点构建出一个DNS实例图.当一个来自某个边缘服务器的实例出现后,部署在同一边缘服务器上的MCCDRM模块将开始向CSPOC模块请求获得该实例与已经存在于DNS实例图中的实例之间的相似性.如果新出现的实例与DNS实例图中的某些实例有相似之处,则将这一实例作为与其最相似的实例的子节点,并对DNS实例图的结构进行更新.同时,将新建立的关联关系发送回CSPOC模块,对相似度测量基准进行更新.如果不存在与新出现的实例具有相似性的实例,则将丢弃这一新出现的实例.构建DNS实例图的具体流程如图3所示.由图3可知,在一个确定的挖掘时间窗口内,15个DNS实例依照时间顺序逐渐被关联到了DNS实例图中.在这15个事例中,Dns_1和Dns_4属于已被各种网络安全设备辨识出的恶意DNS实例.因此,可将它们设为DNS实例图中的根节点.当MCCDRM模块向CSPOC模块请求与Dns_2相关的相似度信息时发现已构建的DNS实例图中没有与其相似的实例,故丢弃Dns_2并等待后续产生的实例.在接下来的实例关联操作过程中,MCCDRM模块继续向CSPOC模块请求相似度信息,逐渐挖掘出与DNS实例图中节点具有相似性的Dns_3,Dns_6,Dns_9,Dns_{10},Dns_{11},Dns_{12}和Dns_{14}.位于图3左侧的上述7个DNS实例逐渐被关联到了右侧的DNS实例图中,变成了可疑DNS实例.从而识别出隐藏在良性DNS活动中的7个可疑DNS响应活动,并构建完成DNS实例图,可用于后续恶意DNS实例挖掘.

2.2.2　恶意DNS实例图构建

在一定的挖掘时间窗口内,筛选出与已检测到的恶意DNS实例相似的可疑DNS实例后,本部分主要负责挖掘出其中新产生的恶意DNS实例.可以提取DNS实例图中呈现出的域名与IP地址间的映射关系,并基于恶意DNS实例与可疑DNS实例指向的

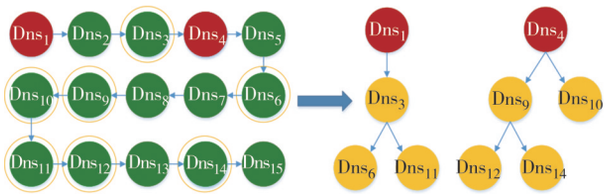


图3　DNS实例图构建过程

Fig.3　DNS instance graph construction process

相同IP的数量实现对两种实例关联性的度量.本文将DNS实例图转化成一种无向域名图,其中DNS实例图中节点所涉及的域名作为节点,这些域名若指向相同的IP地址,则在它们之间建立一条边.通过计算各个IP地址的恶意分值,并将这一分值与域名图的结构相结合,便能度量出各个域名的恶意分值.图4给出了一个由DNS图转换得到域名图的具体实例.图中DNS实例图所涉及的域名和IP地址的映射关系表示为:若Re_Domain1和Re_Domain2映射到了相同的Req_{IP},则在两个域名之间建立一条边.某一域名所映射到的所有IP地址可以形式化表示为式(7)所呈现的多维向量.

$$IP\left(Req_{domain}\right)=\left(Req_{IP_1},Req_{IP_2},\cdots,Req_{IP_n}\right) \tag{7}$$

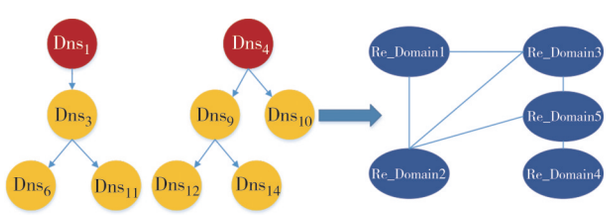


图4　域名图构建过程

Fig.4　Domain graph building process

在式(7)中,有些IP地址是从恶意DNS实例中提取出的,可以设置一个IP地址恶意程度度量指标,用于度量IP地址与恶意DNS实例的关联性,如式(8)所示.通过分析已构建的域名图可知,在映射到某一IP地址的域名中,存在β个域名存在于可疑DNS实例,α个域名存在于恶意DNS实例.由于恶意DNS实例的数量相对较少,为了确保度量值落于一个适当的数值区间内,将α的数值乘以10并增加了一个调节参数δ.与此同时,本文还设置了一个域名恶意程度度量指标,通过将某一域名映射到的所有IP地址的恶意分值做求和运算,便可获得域名的恶意分值,如式(8)和式(9)所示.这一指标可以反映出某一域名是恶意域名的概率,从而实现对恶意DNS实例的有效辨识.

$$\mathrm{MALIP}\left(\mathrm{Req_{IP}}\right)=\begin{cases}\delta\cdot\dfrac{10\alpha-\beta}{\alpha+\beta}, & \dfrac{10\alpha-\beta}{\alpha+\beta}>0\\ 0, & \dfrac{10\alpha-\beta}{\alpha+\beta}\leqslant 0\end{cases} \tag{8}$$

$$\mathrm{MALDomain}\left(\mathrm{Req_{domain}}\right)=\sum_{\mathrm{Req_{IP_i}}\in \mathrm{IP}\left(\mathrm{Req_{domain}}\right)}^{n}\mathrm{MALIP}\left(\mathrm{Req_{IP_i}}\right) \tag{9}$$

如果某一域名的恶意分值超过一定的阈值ϕ，这一域名将被视为恶意域名，并且需对各IP地址的恶意分值进行重新计算.重复以上计算过程，直到没有新的恶意域名产生.将最终挖掘出的恶意域名映射到与之关联的DNS实例，便可实现对恶意DNS实例的有效挖掘.与此同时，与此MCCDRM模块部署在同一边缘服务器上的CSPOC模块也将收到这些挖掘出的恶意DNS实例，并开始收集分析这一恶意DNS响应活动所波及的物联网终端设备中的系统调用信息.

2.3 APT攻击情境中零日攻击活动辨识

零日攻击活动辨识(ZDAARA)方法首先着眼于构造一种系统调用实例图，可用于描述良性系统调用信息和恶意系统调用信息之间的依赖关系.将ZDAARA方法的模块部署在边缘服务器上，用于处理可疑物联网终端设备中的系统调用实例，构建出相应的系统调用实例图.这些可疑的物联终端设备从而被MCCDRM方法挖掘得到，为零日攻击活动辨识过程提供了一种触发机制.接着将系统调用实例图转化为一种网络活动贝叶斯网络，用于描述安全威胁在系统调用实例间的传播演化过程.为了准确客观地描述系统调用实例之间的漏洞和威胁的传播过程以及漏洞修补过程，本文首次提出将安全检测设备的检测精度因素和时间因素加入贝叶斯网络的条件概率表(CPT)中.最终，恶意概率超过一定阈值的系统调用实例将被有效辨识并视为零日攻击活动.如果涉及DNS解析活动，ZDAARA会标记相应的DNS响应活动为恶意，并将信息传递回CSPOC模块进行更新.边缘服务器还将APT攻击路径传输至云数据中心，用于在云数据中心进一步辨识零日攻击活动.

2.3.1 系统调用实例图构建

ZDAARA模块在接收到CSPOC模块传送来的系统调用实例后将提取出这些实例间的依赖关系.本文提出的系统调用实例图构建方法将会在具有相同$\mathrm{File_{Path}}$参数或者$\mathrm{File_{inode}}$参数的实例之间建立一条有向边，方向取决于两个实例出现的时间顺序.如果某一系统调用实例的$\mathrm{Sys_{type}}$参数为“fork”，则该实例将拆分为两个子实例.本文给出了一个由系统调用信息构建系统调用实例图的示例，如图5所示.可以看出9个系统调用实例逐步由系统调用信息构建而成，并形成了一个系统调用实例图.在这9个实例中$\mathrm{Sys_1}$根据在T_1时刻产生的系统调用信息拆分为$\mathrm{Sys_2}$和$\mathrm{Sys_3}$.紧接着，$\mathrm{Sys_2}$对file1进行了“写”操作.如果$\mathrm{Sys_2}$是一个恶意实例，那么file1很有可能已经被其嵌入了安全风险或漏洞.那么，随后与file1发生过交互的$\mathrm{Sys_5}$和$\mathrm{Sys_9}$则很有可能变成恶意攻击活动.因此，可以将$\mathrm{Sys_5}$作为$\mathrm{Sys_2}$的子节点，$\mathrm{Sys_9}$作为$\mathrm{Sys_5}$的子节点逐步关联至系统调用实例图中.在图5左侧的其他实例也将依照同样的关联策略逐步关联至系统调用实例图中.这样，便实现了系统调用实例图的构建，提取了大量系统调用实例间的链式关系.

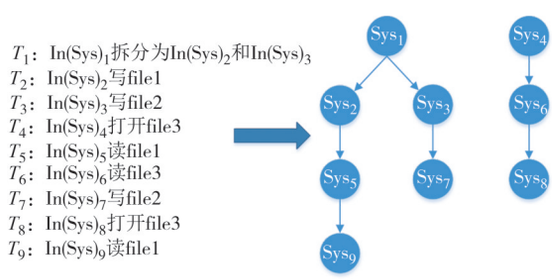


图5 系统调用实例图构建过程

Fig.5 System call instance graph construction process

2.3.2 网络活动贝叶斯图构建

本文将系统调用实例图映射到贝叶斯网络之中，形成了一种可对系统调用实例间影响关系进行描述的网络活动贝叶斯图.当一个父实例$\mathrm{Fa}\left(\mathrm{In(Sys)}_i\right)$已被确认为恶意活动时，其子节点$\mathrm{Sys}_i$则拥有一定的概率会受到恶意风险和漏洞的影响从而也变为恶意活动.基于上述图构建机制，便可构建出对应的网络活动贝叶斯图.

本文给出了一个网络活动贝叶斯图构建实例，如图6所示.图中右侧的网络活动贝叶斯图是由左侧的系统调用实例图转换而构建得到的，实现了对每个系统调用实例可能被辨识为恶意活动的概率的形式化描述.每个系统调用实例的恶意概率主要受两方面因素的影响：Sys自身固有的恶意概率和其父节点$\mathrm{Fa}\left(\mathrm{In(Sys)}_i\right)$的恶意概率.本文为每个系统调用

实例Sys_i添加了一个虚拟节点$Intr(Sys)_i$,用于表示在不考虑父实例影响的情况下每个系统调用实例自身固有的恶意概率.引入一个参数$Prob^{v}_{Sys_i}$用于表示恶意概率随时间推移的递减变化.当某个系统调用实例Sys_i的参数End_{time}与网络安全态势觉察的感知时刻间的时间跨度超过一定时间t_p时,这一漏洞或风险将被安全管理人员修补或控制,因此其恶意概率将下降至“0”.

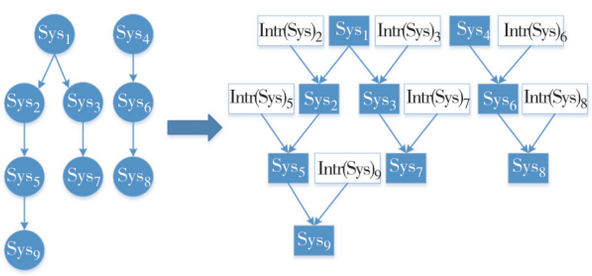


图 6　网络活动贝叶斯图构建过程

Fig. 6　Network Bayesian graph construction process

基于上述机制,在某个网络安全态势觉察时间点将存在4种类型的系统调用实例,表3给出了4种类型在网络活动贝叶斯图中的恶意条件概率.

表 3　系统调用实例的条件概率

Tab.3　Conditional probability for system call instances

指标	Sys_i虚拟节点恶意		Sys_i虚拟节点良性	
	父实例恶意	父实例良性	父实例恶意	父实例良性
恶意条件概率	$Prob_{mm}$	$Prob_{mb}$	$Prob_{bm}$	$Prob_{bb}$

本文给出了每种系统调用实例恶意概率的形式化表示,如式(10)~式(14)所示.以$Prob_{mm}$为例,其恶意概率主要由其固有恶意概率和父实例恶意概率共同影响.由于其父实例具有恶意,当其父实例传播于它的系统漏洞被有效利用后这一实例将受到安全风险的影响,这些漏洞被有效利用的概率可以表示为$Prob^{v}_{ex}$.进而,父实例对于某一系统调用实例恶意概率的影响可以表示为$Prob^{v}_{Fa_Sys_i}$、$Prob^{v}_{ex}$和TPR(真阳性率)3个参数的乘积.系统调用实例自身固有的恶意概率可以表示为$Prob^{v}_{Intr_Sys_i}$与TPR的乘积.$Prob_{mm}$的最终数值则为上述两种恶意概率影响因素的和值.其他类型的恶意概率也可以通过类似的方法进行度量计算.若网络活动贝叶斯图中实例被视为良性实例,有可能是因为安全检测设备存在漏报的现象,则计算其恶意概率时需要将TPR替换为FNR(假阴性率).

$$Prob_{mm} = Prob^{v}_{Fa_Sys_i} \times Prob^{v}_{ex} \times TPR + Prob^{v}_{Intr_Sys_i} \times TPR \tag{10}$$

$$Prob_{mb} = Prob^{v}_{Fa_Sys_i} \times Prob^{v}_{ex} \times FNR + Prob^{v}_{Intr_Sys_i} \times TPR \tag{11}$$

$$Prob_{bm} = Prob^{v}_{Fa_Sys_i} \times Prob^{v}_{ex} \times TPR + Prob^{v}_{Intr_Sys_i} \times FNR \tag{12}$$

$$Prob_{bb} = Prob^{v}_{Fa_Sys_i} \times Prob^{v}_{ex} \times FNR + Prob^{v}_{Intr_Sys_i} \times FNR \tag{13}$$

$$Prob^{v}_{Intr_Sys_i} = \begin{cases} 1 - \dfrac{t - t_{Sys_i}}{t_p}, & t - t_{Sys_i} < t_p \\ 0, & t - t_{Sys_i} \geqslant t_p \end{cases} \tag{14}$$

2.3.3　零日攻击活动辨识

本文给出了在网络活动贝叶斯图中辨识恶意系统调用实例的示例,如图7所示.收到一个新的告警信息时,可立即将网络活动贝叶斯图中对应的系统调用实例标记为恶意实例(见图7中红色方块节点).然后,借助贝叶斯网络的概率推演计算能力可计算得到相关系统调用实例最新的恶意概率.通过计算推演得到各系统调用实例恶意概率,便可将那些超过一定恶意阈值的系统调用实例辨识出来,从而检测到相应的未能被安全检测设备有效辨识的零日攻击活动.如果辨识得到的某个零日攻击活动属于DNS响应活动,那么还需将相关DNS响应信息发送给DNS实例构建模块,用于恶意C&C DNS响应活动挖掘.最终,辨识得到的零日攻击活动即为网络安全态势觉察过程的输出结果,实现了对网络系统中关键恶意活动的有效辨识,可为后续网络安全态势理解过程提供输入数据.

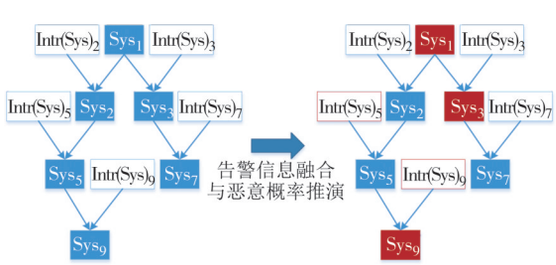


图 7　网络活动贝叶斯图中的恶意系统调用实例辨识

Fig.7　Malicious system call instance identification in Bayesian graph of network activity

3　实验仿真与评估分析

3.1　实验数据获取与实验环境构建

本文采取构建仿真实验环境(如图8所示)并在此环境中模拟3种典型的APT攻击情境来实现对所

提出方法有效性的评估与分析.为了模拟真实物联网系统,本文使用了3台相同配置的主机.其中一台充当云数据中心,另外两台作为受到APT攻击的边缘服务器.边缘服务器1和边缘服务器2上部署了8台虚拟机,充当虚拟物联网终端设备.除此之外,本文还部署了第4台主机充当APT攻击者的角色.这台攻击主机还充当了C&C服务器、Apache服务器和FTP服务器的角色.

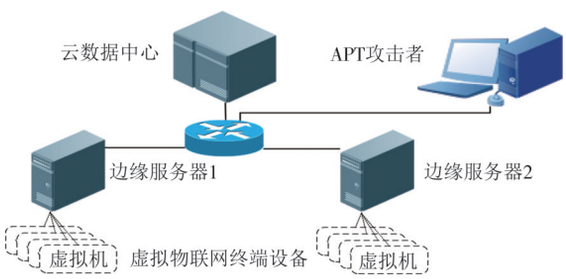


图8 仿真实验环境

Fig. 8 Simulation experiment environment

本文在仿真实验环境中模拟了3种典型的APT攻击情境(Russian Campaign, Op-DeputyDog和SeaDuke).为增加DNS日志数据和系统调用信息的数量,在此仿真系统中发起了一些常规的网络活动.模拟仿真得到的DNS日志和系统调用信息的规模如表4所示.

表4 实验数据集规模表

Tab.4 Table of experiment dataset size

数据类型	数量	大小/kB
DNS日志	630	90
系统调用信息	13 795	29 611

3.2 MCCDRM方法评估

MCCDRM方法的主要作用是挖掘出与恶意C&C服务器进行通信的恶意DNS实例.在获取DNS仿真日志后,可将MCCDRM方法部署在仿真物联网系统中.为验证该方法的有效性,本实验以F_1与FPR作为评估指标参数.两者评估结果结合后进行分析,便可得到该方法辨识效果的整体评估结果.

阈值ϕ是决定某个域名是否会被辨识为恶意域名的关键因素,本实验采取阈值ϕ分别为5、5.5、6、6.5、7、7.5、8、8.5、9时评估MCCDRM方法在3种典型APT攻击情境下的辨识效果,最终得到的评估结果分别如图9和图10所示.依据实验结果可知,随着阈值ϕ的增加,F_1和FPR的值都逐渐减小.通过对两种评估指标参数的实验结果进行比较分析,最终得到当阈值ϕ被设置为7时,效果最优.

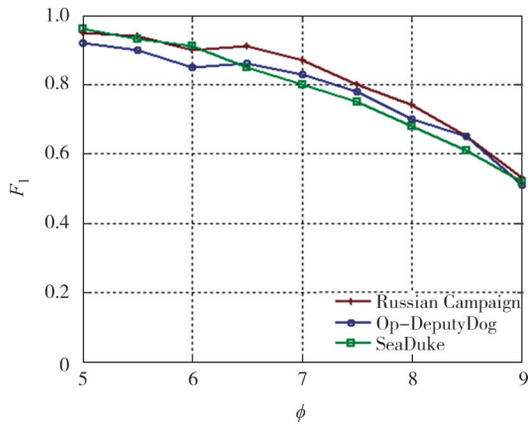


图9 F_1随阈值变化的评估结果

Fig. 9 Evaluation results of F_1 as a function of threshold

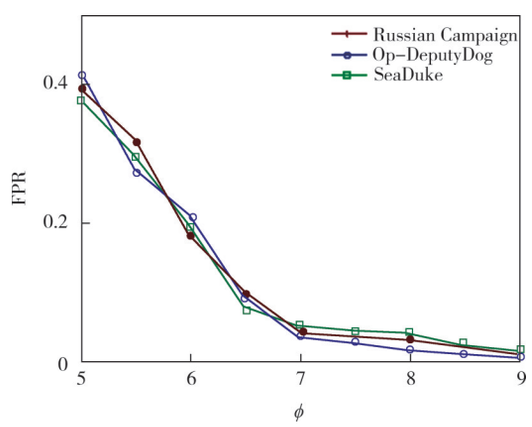


图10 FPR随阈值变化的评估结果

Fig.10 Evaluation results of FPR as a function of threshold

3.3 ZDAARA方法评估

ZDAARA方法的主要作用是辨识隐藏在大规模系统调用信息中的零日攻击活动.本实验仍选取F_1和FPR作为评估ZDAARA方法的指标参数.通过MCCDRM方法提供的APT恶意活动关联分析触发条件,有效减少了ZDAARA方法在物联网系统中的计算开销.为了验证将ZDAARA和MCCDRM相结合是否具有优势,首先在系统调用实例数量变化的情况下对ZDAARA方法进行了评估.本实验对3种典型的APT攻击场景数据运行了ZDAARA方法,评估结果分别如图11和图12所示.

评估结果表明,随着实例数量的增加,F_1和FPR都没有明显的波动.ZDAARA方法单独实施时具有很好的辨识效果,并适应各种规模的系统调用实例规模(该方法对于3种APT攻击情境的F_1值均在0.85以上,且FPR值均未超过0.05).然而,评估过程中发现:随着系统调用实例数量的增加,ZDAARA方法的计算时间也会增加(如图13所示).考虑到物联

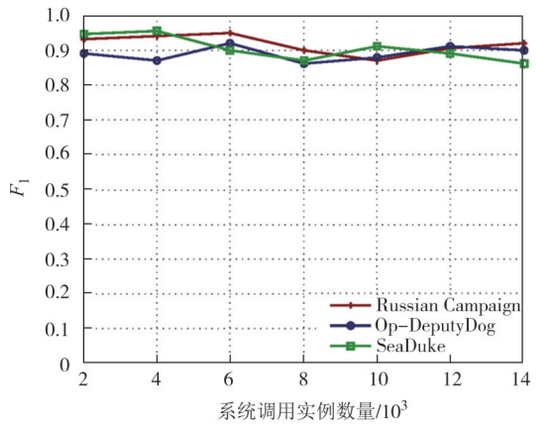


图 11　F_1随系统调用实例数量变化的评估结果

Fig.11　Evaluation results of F_1 as the number of system call instances varies

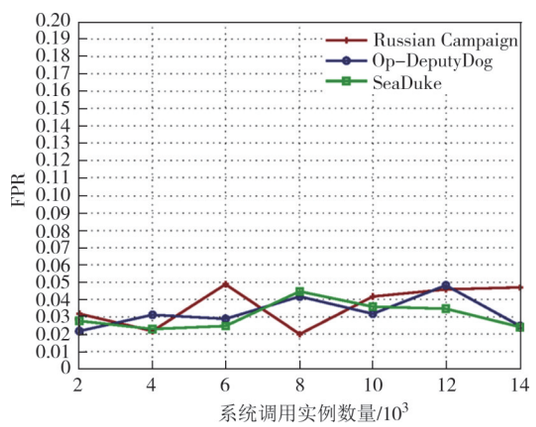


图 12　FPR随系统调用实例数量变化的评估结果

Fig.12　Evaluation results of FPR as the number of system call instances varies

网终端设备和边缘服务器通常在计算资源方面存在限制，因此在真实的物联网环境中使用ZDAARA方法时需要控制数据规模．这证明了本文提出的MCCDRM方法对于将ZDAARA方法应用于物联网系统中起到了不可或缺的作用．

本文实验旨在验证对ZDAARA方法增加运行触发控制机制是否会影响其恶意系统调用实例辨识效果．因此，进行了一个带有和不带有运行触发控制机制的对比评估实验．由于之前的评估结果表明当阈值ϕ设为7时，MCCDRM方法具有较好的恶意DNS辨识效果，在对比实验中便采纳了这一阈值设定．具体的对比结果已呈现在表5中．对比结果表明，当ZDAARA方法由MCDDRM方法触发控制时，其F_1值和FPR值都基本接近于单独运行时的效果，但计算时间实现了超过74%的递减．总体而言，将ZDAARA方法与MCCDRM方法相结合

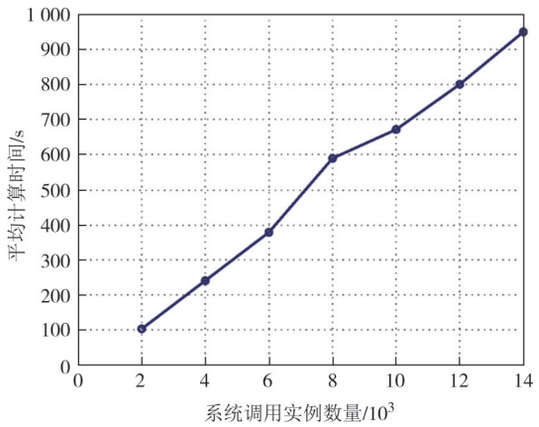


图 13　平均计算时间随系统调用实例数量的变化

Fig.13　The average computation time as a function of the number of system call instances

可以实现在物联网系统中对基于零日攻击的APT攻击活动进行有效辨识．

表 5　有无MCCDRM方法触发的情况下ZDAARA方法的辨识效果对比

Tab.5　Comparison of identification effect of ZDAARA method with and without MCCDRM method trigger

APT攻击情境	ZDAARA单独运行			ZDAARA由MCCDRM触发		
	F_1	FPR	计算时间/s	F_1	FPR	计算时间/s
Russian Campaign	0.91	0.047	952	0.87	0.041	246
Op-DeputyDog	0.90	0.025	952	0.86	0.037	246
SeaDuke	0.86	0.024	952	0.83	0.019	246

3.4　对比实验分析

本文进一步对比了APTIZDM与3种典型的APT检测方法(TerminAPTor[10]、Spearphishing-based[11]和context-based[12])的各自优势．TerminAPTor重点关注通过跟踪信息流来检测APT的活动．通过分析如表6所示的对比结果可知，作为对比组的3种APT检测方法很难处理好F_1值和FPR之间的平衡．其中唯一能与APTIZDM具有较为相似检测效果的context-based方法，与本文提出的方法相比具有无法实现实时APT攻击活动检测的弱点．

表 6　APTIZDM与其他APT检测方法的对比结果

Tab.6　Comparison results between APTIZDM and other APT detection methods

APT检测方法	检测速度	F_1	FPR
APTIZDM	实时	0.85	0.032
TerminAPTor	实时	0.96	0.247
Spearphishing-based	实时	0.93	0.151
context-based	离线	0.77	0.024

4 结 论

本文提出了一种对APT攻击情境中零日攻击有效辨识的网络安全态势觉察方法.为了应对持续实时挖掘恶意活动会消耗大量计算资源的挑战,设计了一种MCCDRM方法用于控制APT恶意活动关联分析触发条件.本文还提出了ZDAARA方法用于辨识IDS无法检测到的零日攻击活动,解决了传统网络态势觉察方法应用于物联网系统时所面临的挑战.实验结果表明,将MCCDRM与ZDAARA方法相结合可以有效辨识物联网系统中基于零日漏洞的APT攻击活动,具有较高的辨识准确度和可以接受的误报率.

参考文献

[1] ZHENG K N, ZHANG Z J, CHEN Y, et al. Blockchain adoption for information sharing: risk decision-making in spacecraft supply chain[J]. Enterprise Information Systems, 2021, 15(8): 1070-1091.

[2] TUBÍO FIGUEIRA P, LÓPEZ BRAVO C, RIVAS LÓPEZ J L. Improving information security risk analysis by including threat-occurrence predictive models[J]. Computers & Security, 2020, 88: 101609.

[3] KHALID M N A, AL-KADHIMI A A, SINGH M M. Recent developments in game-theory approaches for the detection and defense against advanced persistent threats (APTs): a systematic review[J]. Mathematics, 2023, 11(6): 1353.

[4] SOUSA D X, CANUTO S, GONÇALVES M A, et al. Risk-sensitive learning to rank with evolutionary multi-objective feature selection[J]. ACM Transactions on Information Systems, 2019, 37(2): 24.

[5] ZHANG D D, USA L U, NAULT B R. Measuring risk from IT initiatives using implied volatility[J]. Journal of the Association for Information Systems, 2019, 20(11): 1566-1588.

[6] SHARMA A, GUPTA B B, SINGH A K, et al. A novel approach for detection of APT malware using multi-dimensional hybrid Bayesian belief network[J]. International Journal of Information Security, 2023, 22(1): 119-135.

[7] VICE V. NIST vs. ISO: What's the Difference?[EB/OL].(2023-04-24)[2023-11-16]. https://www.auditboard.com/blog/nist-vs-iso-whats-the-difference.

[8] SÁNCHEZ-ZAS C, VILLAGRÁ V A, VEGA-BARBAS M, et al. Ontology-based approach to real-time risk management and cyber-situational awareness [J]. Future Generation Computer Systems, 2023, 141(C): 462-472.

[9] WANG Y, LIAN X, CHEN L. Efficient SimRank tracking in dynamic graphs[C]//2018 IEEE 34th International Conference on Data Engineering (ICDE). Paris, France: IEEE, 2018: 545-556.

[10] HAN W J, XUE J F, WANG Y, et al. APTMalInsight: identify and cognize APT malware based on system call information and ontology knowledge framework[J]. Information Sciences, 2021, 546: 633-664.

[11] ABU TALIB M, NASIR Q, BOU NASSIF A, et al. APT beaconing detection: a systematic review [J]. Computers & Security, 2022, 122: 102875.

[12] FOMICHEV M, ABANTO-LEON L F, STIEGLER M, et al. Next2You: robust copresence detection based on channel state information[J]. ACM Transactions on Internet of Things, 2022, 3(2): 1-31.