计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.190700199

基于流量指纹的物联网设备识别方法和物联网安全模型

杨威超[1,2]　郭渊博[1]　李　涛[1]　朱本全[2]
1 信息工程大学密码工程学院　郑州 450000
2 61213 部队　山西 临汾 041000
(79579@163.com)

摘　要　物联网(Internet of Things,IoT)的大规模部署应用,使得有漏洞的物联网设备也可能联入网中。攻击者利用有漏洞的设备接入目标内部网络,就可潜伏伺机发起进一步的攻击。为防范这类攻击,需要开发一种对可疑设备接入控制并管理内部设备的安全机制。首先,为实现对可疑设备的接入控制,文中给出了一种设备识别方法,通过设置白名单,构建通信流量特征指纹,使用随机森林方法来训练设备识别模型;其次,为管理内部设备,提出了一种智能安全管理模型,构建基于资产、漏洞、安全机制等的本体威胁模型;最后,通过实验验证了设备识别模型的检测效果,其识别准确率达到96%以上,并将其与已有类似方法进行对比,结果证明了所提方法具有更好的检测稳定性。

关键词:本体威胁建模;物联网设备识别;流量特征提取;白名单;随机森林
中图法分类号　TP393

Method Based on Traffic Fingerprint for IoT Device Identification and IoT Security Model

YANG Wei-chao[1,2],GUO Yuan-bo[1],LI Tao[1] and ZHU Ben-quan[2]
1 School of Cryptography,University of Information Engineering,Zhengzhou 450000,China
2 61213 Troops of the Chinese People's Liberation Army,Linfen,Shanxi 041000,China

Abstract　The large-scale deployment of the Internet of Things makes it possible for vulnerable IoT devices to be connected to the network. When an attacker uses a vulnerable device to access the target internal network,it can lurk to wait for an attack. To prevent such attacks,it is necessary to develop a security mechanism for access control of suspicious devices and management of internal devices. Firstly,in order to realize the access control of suspicious devices,a device identification method is given in this paper. By setting a white list,a communication traffic feature fingerprint is constructed,and a random forest method is used to train the device identification model. Secondly,to manage internal devices,an intelligent security management model is proposed to build an ontology threat model based on assets,vulnerabilities and security mechanisms. Finally,the experimental results verify the detection performance of the device recognition model,the recognition accuracy rate is above 96%. Compared with the existing similar methods,the results prove that the proposed method has better detection stability.

Keywords　Ontology threat modeling,IoT device identification,Traffic feature extraction,White list,Random forest

1　引言

物联网数量的激增是未来的趋势。根据最近的预测,物联网设备将按市场需求呈指数级增长,2030年物联网设备的数量将达到1250亿[1]。

海量物联网设备的使用和其应用技术的普及方便了我们的生活[2],但其在服务、技术、设备和协议(如无线、有线、卫星、蜂窝和蓝牙等)上的异构性使得物联网的管理愈加复杂[3-4]。由于很多智能设备的生产供应商都是不具备网络安全专业知识的传统家用电器制造商,因此很多设备先天存在漏洞。攻击者利用有漏洞的设备接入目标网络,潜伏伺机发起攻击,从而导致目标网络面临严重的安全威胁[5]。

在大型的公司或者组织内部,对有漏洞设备进行接入控制以及内部安全管理尤为重要。

文献[6]发现公司内部员工倾向于将大量物联网设备连接到家庭网络,而25%～50%的员工表示已经将这些物联网设备中的某一个连接到了企业网络。这些通过员工连接到家庭或者企业内部网络的设备,成为了攻击者发动攻击的据点,比如安装在会议室的智能电视。文献[7]指出,可以使用Skype应用程序来提升一个小部件的权限,使其拍摄想要的图像,并将图像信息泄露给远程FTP服务器。文献[8]介绍了另一种攻击类型,虽然显示器已关闭,但植入的恶意软件仍然能够捕获周围的声音,并通过WiFi将其非法传输给第三方。面对这些种类繁杂、路径多样的攻击,各种组织的内部网

到稿日期:2019-07-29　返修日期:2019-11-27　本文已加入开放科学计划(OSID),请扫描上方二维码获取补充信息。
基金项目:信息保障技术重点实验室基金(614211203010417)
This work was supported by Foundation of Science and Technology on Information Assurance Laboratory (614211203010417).
通信作者:郭渊博(yuanbo_g@hotmail.com)

络应该重新考虑是否允许这些设备轻易连接网络，并考虑如何对连接到内部网络的设备进行管理。

为解决上述问题，需要在设备接入网络时以及接入网络后部署对应的安全策略。对此，本文提出了基于流量特征的设备识别接入控制技术及基于本体威胁建模的智能安全管理模型。

本研究的主要贡献如下：

(1)设计了一种通过基于流量特征提取的设备识别方法，从而实现了对非白名单设备的接入控制；

(2)设计了一种智能化的安全管理模型，并运用了本体威胁建模的框架对内网设备进行安全管理；

(3)实验验证了通过提取设备初始设置阶段的通信流量特征，运用随机森林方法来进行设备自动识别的模型，识别准确率在96%以上。

本文第2节主要介绍与物联网相关的设备识别技术、异常检测技术及安全本体构建技术；第3节介绍设备识别方法和安全管理模型的框架结构；第4节通过实验验证设备识别的效果，并对实验结果进行评估；最后总结全文并进行展望。

2 相关工作

2.1 设备识别

传统的设备识别技术在无线通信中被广泛应用，早期的无线通信指纹工作主要用于识别基于硬件和驱动的设备的特性[9-10]。面向物联网的设备识别技术主要利用与传感器相关的特征来唯一识别设备[11-12]。由于大量廉价的物联网设备使用的内部驱动或者硬件等大致相同，而且并不是所有的设备都配置传感器，因此这些已有方法不能用来准确识别物联网的设备型号。

本文利用机器学习技术，根据设备流量特征构建指纹，并训练分类器来识别设备。已有的利用流量特征开展设备识别的工作有许多。文献[13-14]分别采用了大量的特征和复杂的信号处理技术，这些方法需要设备具有较强的计算能力，这给资源受限的物联网设备带来了挑战。而本文采用设备通信设置阶段流量的少量特征，一次性地识别设备的具体型号，对资源要求较低。Meidan等[15]提出了在网络流量数据中应用机器学习算法来准确识别连接到网络的物联网设备，但没有将设备识别与网络安全威胁应对措施结合。而本文设法将设备型号与异常检测和威胁建模结合起来，构建一种安全管理机制。Shaikh等[16]提出了一种使用机器学习识别发现企业网络中存在恶意行为的物联网设备模型。该方法能识别被感染的设备，但没有对设备型号进行分类。Salman等[17]针对物联网的异构性和大规模部署，提出了一种智能集成在网络边缘的模型。Thangavelu等[18]提出了一种分布式的指纹解决方案DEFT，其解决了智能家居和企业网络中常见设备的识别问题。以上方案并没有对设备识别以及发现异常后的响应机制做详细的阐述，而本文详细阐述了基于设备型号的异常检测机制以及基于知识图谱的威胁响应模型。

Miettinen等[19]提出了一种自动设备型号识别模型Sentinel，该模型首先识别新接入网络的设备，然后对设备进行漏洞评估，最后根据评估结果限制设备的通信。该研究在设备识别部分根据提取设备接入网络时的通信设置流量特征，通过随机森林的分类方法进行设备识别。本文在该研究的基础上对特征的选取以及指纹的构建方法做了一些调整，提高了设备识别的准确率。

2.2 物联网安全模型

关于物联网，研究者们提出了一些异常检测方法和安全管理框架。

分布式异常检测模型在物联网安全领域得到了广泛应用。Diro等[20]将基于深度学习的分布式的异常检测架构与中心式的异常检测架构进行对比，进一步证明了前者具有更加优越的性能。借助安全网关构建分布式异常检测架构是一种更好的选择。文献[14]就设计了基于设备型号的分布式异常检测模型，但是其识别的设备型号是抽象的，可解释性较差；并且该模型只能检测基于行为模式的异常，不能够在识别设备型号后发现设备可被利用的漏洞，从而提前采取一些安全措施。而本文在设备型号识别后，会主动收集设备的相关信息，并构建基于知识图谱的威胁模型，帮助安全管理人员提前感知威胁。与本文思路较为相似的文献[21]中，作者提出了一种基于角色的异常检测方法和态势感知模型，用于解决智能楼宇系统中的安全问题。该方法基于BACnet协议的语义信息来定义角色并进行异常检测，需要大量的专家知识，可扩展性不强。本文构建基于设备型号信息的威胁模型，不受具体某种协议的限制，通过构建设备信息的知识图谱来实现威胁的自动感知。

传统的安全管理方法有很多[5,22-24]，这些方法采用威胁建模的机制进行安全管理，但是需要人工构建威胁模型，面对大规模的数据时无法进行有效的处理，并且不能实现智能推理和识别，不善于发现隐含的威胁。而知识图谱的方法可以实现威胁的智能化管理，已经在物联网安全领域得到了一些研究。文献[25-26]分别介绍了本体构建方法和基于本体的安全框架，利用知识图谱的知识推理功能为发现的威胁提供相应的安全服务。而本文侧重于构建基于设备信息和异常警报的知识模型来发现漏洞和威胁。

3 物联网内部威胁描述

本文重点研究物联网内设备安全性问题带来的内部威胁的情形。与由大型僵尸网络执行的DDoS攻击不同，这种威胁在于人员违反内部网络安全策略，通过有目的或无目的地擅自将设备接入网络，使得设备中的恶意软件感染内部网络的其他设备，而攻击者能利用设备本身的漏洞进行高级攻击。

设备接入网络时，大致可以定义两种威胁场景。1)未定向攻击：物联网设备在接入网络前已经被恶意软件感染，这些恶意软件是为了感染尽可能多的联网设备而不针对某个特定的网络；2)定向攻击：攻击者故意将恶意软件植入物联网设备，这个设备可能会连接到某个特定的内部网络，如供应链攻击，即物联网设备在到达用户手中之前的生产、分发和处理过程中被植入后门。此外，也可能是攻击者提前获得了对某个特定用户设备的控制权，当该用户违反安全策略擅自将设备接入内部网络时，攻击者通过该设备进行内部网络相关敏感信息的非法获取等攻击。

设备接入网络后，假设连接到内部网络的设备存在漏洞，并可被攻击者利用。攻击者利用漏洞，可以进行以下操作：1)泄露或窃取敏感数据；2)非法接入；3)在用户网络中注入错

误的或者篡改的信息；4）流量嗅探。攻击模型如图 1 所示。

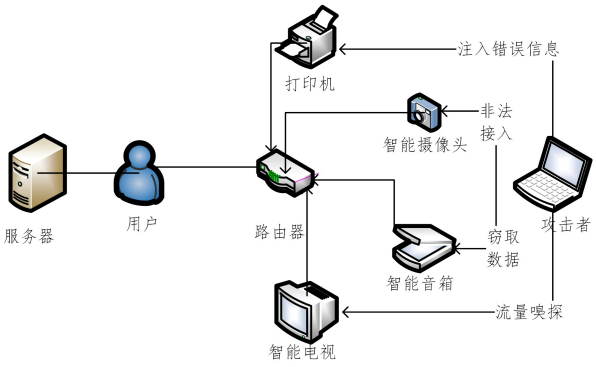

图 1　威胁模型

Fig. 1　Threaten model

4　方法模型

针对上述安全威胁，我们采用设备识别系统对白名单之外的设备进行接入控制，并采用安全管理系统对白名单内的设备部署进行通信限制等。

通过白名单对设备进行接入控制，可以避免内部网络接入大量含有高危漏洞的设备。同时，即使没有高危漏洞的设备被定向攻击者植入后门，设备流量通信特征也可能发生变化，从而不能通过系统的设备识别接入验证。

即使未定向攻击或者定向攻击的设备侥幸接入到内部网络，安全管理系统也会对白名单内部设备实施持续的流量监测以及基于本体的威胁建模，进而根据监测和建模结果部署对应的安全管理措施。

4.1　设备识别模型

为了避免存在严重漏洞隐患的设备接入内部网络，实现对非法设备的接入控制，将威胁挡在内部网络之外，可通过维护一个设备白名单列表来限制非白名单设备接入网络。设备识别的过程如图 2 所示，具体步骤为：流量包捕获、特征提取、指纹构建、模型训练和设备识别。

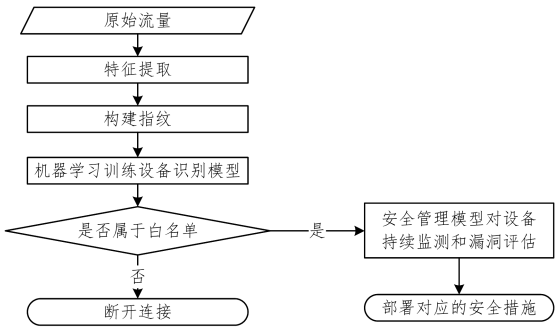

图 2　设备接入控制和内部管理流程图

Fig. 2　Device access control and internal management flow chart

4.1.1　白名单

当被感染的设备连接到内部网络后，攻击者能够发动类似于第 2 节中所描述的多种形式的攻击。为了缓解此类安全威胁，可以有意识地控制设备连接内部网络，比如拒绝存在明显可被利用的漏洞的设备接入网络。在小型的内部组织中维护一个白名单是容易的，安全管理人员经过培训后可以对连接到内网的设备定期进行巡查。但是在大型的内部组织网络中，人员组成的复杂性与设备的多样性，使得监控未经授权的连接设备变得很困难。这就需要构建一种自动化的设备识别方法，防止白名单以外的设备连接内部网络而带来安全威胁。

基于白名单的设备识别系统与安全事件管理模型相关联，对白名单之外的设备进行接入控制并对白名单内的设备部署进行通信限制等，如图 2 所示。设备识别系统通过不间断或者周期性地监视内网设备，可以发现某个时刻、某个地方的某个未经授权的设备尝试非法接入内部网络。

虽然黑名单同样可以实现对某类非法设备的接入控制，比如电子邮件中的垃圾邮件过滤，但是考虑到物联网设备型号以及不同设备本身携带的漏洞的复杂多样性，维护一个白名单的设备列表比维护一个黑名单的设备列表容易得多。较短的白名单设备列表有助于提高基于机器学习的设备识别系统的检测效率。此外，收集白名单中设备型号对应的流量特征数据比黑名单容易得多，因为系统不可能收集所有黑名单设备的流量特征，而是通过建立白名单设备清单的流量特征库，并将新接入内网的设备型号的流量特征与已有特征库进行对比，如果发现设备不属于白名单，则将断开该设备与内网的连接。

为了防止设备伪装成白名单中的某个设备型号企图绕过设备识别机制，我们采取持续监视的安全策略，周期性地采集不同 IP 数据流对应的流量特征，最大限度地降低对未授权设备的漏报率。但是由于目前缺乏公开的实验数据集，我们只对设备连接入内部网络时的通信流量进行了特征提取与分析，并通过实验验证了其设备型号的识别率。

4.1.2　特征提取与指纹构建

本文的指纹构建基于被动式的流量检测。当智能设备第一次连接到网络与网关通信时，该设备会遵循设备或者供应商特有的设置流程进行通信。不同设备设置流程中的通信序列具有差异性，这就是本文获取指纹的来源。

当一个新的设备与网关通信时，网关记录该设备设置阶段的 n 个通信数据流量包：

$$pkt\_sequence=\{p_1,p_2,p_3,\cdots,p_n\} \tag{1}$$

本文采用的流量数据格式为原始 pcap 文件，从该 pcap 文件中提取了 20 个特征，这与文献[19]中给出的 23 个特征不同。去掉一些通信设置中并不能充分区分设备的特征 IP options 和 packet content 中的 raw data 选项，最后的特征如表 1 所列。

表 1　20 个特征类型及对应特征变换方法

Table 1　20 Feature types and corresponding feature transformation methods

原始特征类型	特征变换
链路层协议(2)：ARP/LLC	存在与否：0，1 变换
网络层协议(4)：IP/ICMP/ICMPV6/EAPoL	存在与否：0，1 变换
传输层协议(2)：TCP/UDP	存在与否：0，1 变换
应用层协议(8)：HTTP/HTTPS/DHCP/BOOTP SSDP/DNS/MDNS/NTP	存在与否：0，1 变换
包长度(1)	以 50 字节为步长，将包长度转换为 0～8 的数字
IP 地址(1)	转换为地址变化频次
源/目的端口号(2)	根据端口号大小将端口号转换为 0～3 的数字

每个包提取 20 个特征组成向量：

$$\boldsymbol{p}_i=\{f_{1,i},f_{2,i},f_{3,i},\cdots,f_{n,i}\},i\{1,\cdots,n\} \tag{2}$$

每个设备的指纹就是 n 个包的特征向量组成的 $20*n$ 的矩阵：

$$\boldsymbol{Features}=\begin{bmatrix} f_{1,1} & f_{1,2} & \cdots & f_{1,n} \\ f_{2,1} & f_{2,2} & \cdots & f_{2,n} \\ \vdots & \vdots & & \vdots \\ f_{20,1} & f_{20,2} & \cdots & f_{20,n} \end{bmatrix} \tag{3}$$

由于本文提取的特征为包头信息，因此这样的方法可以用于加密流量。本文在对提取的特征进行处理时借鉴了文献[19]中给出的方法，首先对协议存在与否进行 0,1 建模，这样每一种协议都最终形成一维特征向量：

$$\boldsymbol{protocol}=\{f_1,f_2,\cdots,f_n\},f_i\in\{0,1\} \tag{4}$$

针对端口地址号，采用端口区间建模法：

$$\begin{Bmatrix} None\Rightarrow f=0; \\ [0,1\,023]\Rightarrow f=1; \\ [1\,024,49\,151]\Rightarrow f=2; \\ [49\,152,65\,535]\Rightarrow f=3 \end{Bmatrix} \tag{5}$$

得到特征向量：

$$\boldsymbol{port}=\{f_1,f_2,\cdots,f_n\},f_i\in\{0,3\} \tag{6}$$

与文献[19]不同，本文对 IP 地址不再采用计算 IP 地址个数的方法构建指纹，因为设备设置阶段的很多数据包没有目的地址，这样的特征处理方法不能很好地体现设备通信设置的差异性。本文采用计算 IP 地址变化频次的方法来构建指纹，这样就可以形成 IP 地址变化频次的特征向量：

$$\boldsymbol{ip\_frequency}=\{0,1,1,2,2,2,3,\cdots,i,i+1,\cdots,n\} \tag{7}$$

这样可以更加准确地描绘设备通信设置阶段 IP 地址变化的特点，这也是本文方法能取得较高设备识别准确率的一个关键点。

对于包长度的处理，文献[19]直接将包字节大小统计出来作为特征；本文对包长度同样做了建模：

$$\begin{Bmatrix} [0,50]\Rightarrow f=0;[51,100]\Rightarrow f=1; \\ [101,150]\Rightarrow f=2;[151,200]\Rightarrow f=3; \\ [201,250]\Rightarrow f=4;[251,300]\Rightarrow f=5; \\ [301,350]\Rightarrow f=6;[351,400]\Rightarrow f=7; \\ [401,+\infty]\Rightarrow f=8; \end{Bmatrix} \tag{8}$$

通过对包长度的转化，可以形成一个特征向量：

$$\boldsymbol{pkt\_len}=\{f_1,f_2,\cdots,f_n\},f_i\in\{0,8\} \tag{9}$$

在对特征值进行转化后，特征向量就组成了特征矩阵。

由于特征指纹属于高维数据，因此本文选用能够快速处理高维数据并且不容易产生过拟合的随机森林的方法训练设备识别模型。但是矩阵形式的指纹无法轻易使用随机森林算法，遂将特征矩阵的行向量拼接，得到一个一维行向量：

$$\boldsymbol{f\_row}=\{f_{1,1},f_{1,2},\cdots,f_{1,20},f_{2,1},f_{2,2},\cdots,f_{20,n-1},f_{20-n}\} \tag{10}$$

在指纹中，n 的取值要兼顾设备识别的准确度和设备识别的速度。n 的取值较大，特征较多，识别准确度就会提高，但抓取的数据包过多会影响设备识别的速度。根据后面实验结果，选取 $n=12$ 作为数据包的数量。这样就形成了设备一次通信设置过程中的指纹向量——一个 240 维的行向量：

$$\boldsymbol{f\_row\_}12=\{f_{1,1},\cdots,f_{1,20},f_{2,1},\cdots,f_{20,12}\} \tag{11}$$

经过以上步骤，最终将原始 pcap 文件转化为 csv 形式的特征数据集。

4.1.3 识别检测算法

为了对训练集中设备型号的特征指纹集进行学习，建立模型并利用测试集评估训练效果，本文用有监督学习的随机森林算法来建立检测模型。

随机森林算法由 Leo 和 Adele 最先提出，该算法结合了 bagging 和由 Ho 和 Amit 提出的选择随机属性的思想，是一个由多棵决策树分类器$\{h(x,\boldsymbol{\theta}_k)\}$组成的集成学习分类器。其中，$\{\boldsymbol{\theta}_k\}$是独立同分布的随机向量，$k$ 表示决策树分类器的数量，单个决策树分类器根据输入的测试样本集 x 产生分类结果，最终通过投票确定测试样本类别。随机森林算法的流程如图 3 所示。

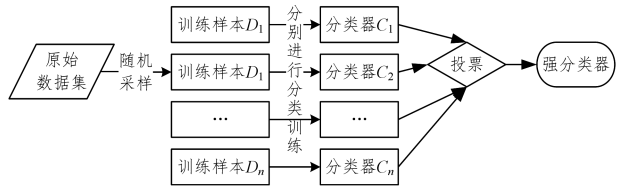

图 3 随机森林算法流程示意图

Fig. 3 Schematic diagram of random forest algorithm flow

随机森林算法有很多优点：首先，它是一种集成学习算法，通过组合若干单个分类器的分类结果，对测试样本进行分类，相比单个分类器具有更好的分类效果和泛化能力；由于特征子集是随机选取的，因此该算法能够处理高维度数据，且不必做特征选择；该算法的训练过程中决策树之间相互独立，训练速度快。

4.2 安全管理模型

在对接入网络的设备型号进行识别后，如果发现设备型号不属于白名单，则断开该设备的连接，以保护内网的安全；如果该设备型号属于白名单，该设备同样可能具有已经被发现的或者还未被发现的漏洞，为了充分保证内网的安全，我们结合云服务器中收集的信息分析设备可能存在的漏洞，进而对该设备采取一些通信限制和异常检测等措施。

为了防止白名单中设备漏洞被攻击者利用，从而对内网安全构成威胁，本文设计了一种智能化的设备识别、异常检测与威胁感知模型，如图 4 所示。

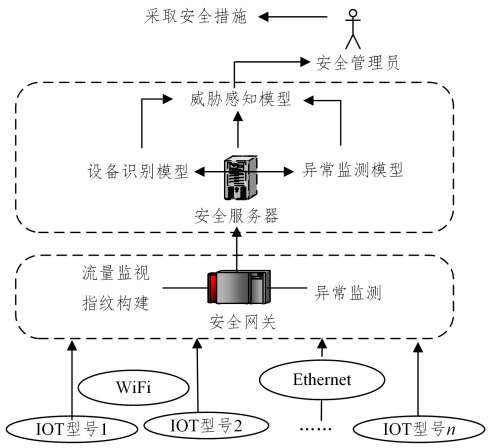

图 4 IOT 安全管理系统模型

Fig. 4 IOT security management system model

本模型旨在根据物联网设备型号类型和异常报警信息实现威胁与安全措施的智能关联，帮助管理人员限制相应设备的某些功能，使得攻击者不可以利用设备漏洞发起攻击。这样，即使带有漏洞的设备连接到网络中，安全模型也可以有效降低其潜在的攻击威胁和影响。

本模型基于上一节提供的自动化的设备识别方法，在设备刚接入网络时自动识别设备的类型，根据白名单对设备型号进行筛选；其次，在白名单内的设备运行过程中进行实时的异常监测，及时产生报警信息；最后，威胁模型通过云服务器提供的信息对设备型号进行漏洞评估，并关联对应的通信限制等安全措施，帮助安全管理人员消除威胁。

具体的通信限制措施如下：1)严格隔离，即仅允许设备与其他被定义为严格隔离的设备进行通信，这些设备都不能连接互联网；2)受限隔离，即仅允许设备与其他受限隔离的设备通信，且只允许其与互联网上的有限个目标(比如供应商的云服务)进行通信；3)可信隔离，即允许设备与其他可信隔离设备进行通信，并且不受限制地访问互联网络。

该模型包含安全网关和安全服务器。安全网关负责监视设备，获取流量，构建指纹以及检测设备异常。安全服务器根据安全网关提供的流量、指纹和异常检测结果识别设备类型，构建异常检测模型，以及完成安全信息知识库的关联。

4.2.1 安全网关

安全网关充当互联网的本地访问网关，物联网设备通过WiFi或以太网与之连接。安全网关除了承担网关路由器的功能外，还具备以下功能：首先，采集物联网设备识别所需的通信流量，并从流量中提取用于进行设备型号识别的指纹；其次，持续监视设备流量的变化，检测因攻击行为而产生的流量模式异常。安全网关同样为安全服务器传输指纹、警报等数据供安全服务器训练设备识别模型、异常检测模型以及基于前两者的威胁感知模型。

4.2.2 安全服务器

安全服务器与安全网关相互支撑。安全服务器包含3个部分：设备识别模型、异常检测模型以及威胁关联模型。设备识别模型采用机器学习的方法，根据安全网关构建的指纹训练设备识别分类器。异常检测模型采用基于设备型号的方法，在完成设备类型的识别后，安全服务器传递给安全网关基于设备型号的异常检测模型，在收到基于特定设备型号的异常检测模型后，安全网关开始持续监测该设备的流量，并在发现模式异常后及时预警。威胁关联模型根据设备型号和报警信息识别并分类各种威胁，并与知识库中相应的安全服务相关联，为安全管理人员推送最恰当的安全管理方案。

4.2.3 基于本体的威胁感知模型

为了及时发现内部网络中的设备漏洞，智能化部署对应的通信限制等安全措施，我们构建了自动化的威胁感知模型。在此，我们对基于本体的威胁建模进行了研究设计，通过本体将设备型号即资产与漏洞、威胁和安全机制等相互关联起来。

一方面，在设备型号得到识别后，威胁感知模型根据设备型号对应的漏洞库预知潜在威胁，帮助安全管理人员采取通信限制等措施以事前预防风险；另一方面，在收到异常检测系统报警后，威胁感知模型能够根据报警信息及时反应，分析应对威胁，帮助安全管理人员事后处理风险。

本体是建模的一种工具。目前已存在一些关于安全本体的研究[26-29]，不同研究中安全本体涵盖的范畴不同，有的侧重于一般概念，有的针对特定领域。构建包含资产、漏洞、警报、威胁、安全机制和关系的物联网安全本体，可以帮助安全管理人员管理风险。本体模型如图5所示。

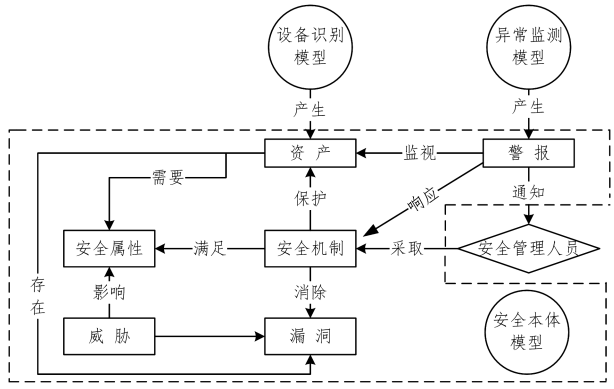

图5 安全本体模型

Fig. 5 Security ontology model

资产作为本体中的一种抽象概念，包含物理资产即设备、设备上的软件，以及设备使用的通信方式如WiFi和Ethernet等。资产要求安全属性被认为是安全的，例如可用性、机密性和完整性等。每个威胁都会影响一个或多个安全属性，安全机制可以满足这些安全属性。

漏洞是设备或软件中的缺陷，当它们被发现时，供应商一般会发布补丁来修复。但对于物联网设备来说，打补丁往往不及时，所以需要使用安全机制来预防攻击者利用被发现的漏洞。

安全机制描述保护已知漏洞的工具和方法。选择安全机制最重要的方面在于资产的宝贵水平，根据成本等因素来实施恰当的保护。安全机制一般包括侦探、预防、纠正、恢复、响应等。

威胁表示利用设备和软件等的漏洞弱点可以发起的攻击，一般利用一个或多个漏洞。威胁等级代表对资产造成损害的程度，包括数据泄露和数据破坏等问题。

警报代表异常检测模型感知到的威胁类型，在设备通信有异常行为时，安全机制会做出响应，比如提高安全保护等级，限制设备通信等。

4.3 小结

本节介绍了针对控制非白名单设备接入的设备识别方法设计以及针对白名单设备在内部网络中管理的安全管理模型设计。

综上可以看出，设备识别是两种方法模型的设计基础。保持设备识别的高准确率，对于基于设备型号的异常检测以及威胁建模至关重要。同时，考虑到数据集等实验条件的限制，下一节只对设备识别过程进行部分实验验证与评估。

5 实验与评估

5.1 数据集

本文采用的数据集来自公开数据集[30]，用于实验的设备都是市场上消费者常用的有代表性的设备，包括智能照明、家

庭自动化、安全摄像头、家用电器和健康监测设备等共 31 种。大多数测试设备都是通过 WiFi 或者以太网连接用户网络，但有少数设备利用其他物联网协议（如 ZigBee，Z-Wave）通过集线器设备间接连接网络，此类设备的流量从集线器设备间接生成。设备列表如表 2 所列。

表 2 实验数据集设备类型

Table 2 Experimental data set device type

序号	设备类型	通信方式
1	Aria	WiFi
2	HomeMaticPlug	其它
3	Withings	WiFi
4	MAXGateway	以太网
5	HueBridge	ZigBee、以太网
6	HueSwitch	ZigBee
7	EdnetGateway	WiFi
8	EdnetCam1	WiFi、以太网
9	EdnetCam2	WiFi、以太网
10	EdimaxCam1	WiFi、以太网
11	EdimaxCam2	WiFi、以太网
12	Lightify	WiFi、ZigBee
13	WeMoInsightSwitch1	WiFi
14	WeMoInsightSwitch2	WiFi
15	WeMoLink	WiFi、ZigBee
16	WeMoswitch1	WiFi
17	WeMoswitch2	WiFi
18	D-LinkHomeHub	WiFi、以太网
19	D-LinkDoorSensor	Z-Wave
20	D-LinkDayCam	WiFi、以太网
21	D-LinkCam	WiFi
22	D-LinkSwitch	WiFi
23	D-LinkWaterSensor	WiFi
24	D-LinkSiren	WiFi
25	D-LinkSensor	WiFi
26	TP-LinkPlugHS110	WiFi
27	TP-LinkPlugHS100	WiFi
28	EdimaxPlug1101W	WiFi
29	EdimaxPlug2101W	WiFi
30	SmarterCoffee	WiFi
31	iKettle2	WiFi

为了使每一个被测试的设备能够产生足够的训练数据，将其设置过程重复进行 20 次，即每次初始化设置完成后，设备会被强制恢复出厂设置。初始化设置通常包括：激活设备，借助供应商提供的应用软件将设备连接到 WiFi 或者以太网络，将 WiFi 凭证传输到用户网络等。

5.2 实验设置及评估参数

5.2.1 实验环境

本文异常检测实验的硬件环境为：操作系统为 Windows10，内存为 16GB，Intel Core i7-4790 CPU@3.60GHz；所用的软件环境为 Pycharm2017。

5.2.2 评估方法

本文主要利用表 3 所列评估参数来评估实验结果。通过混淆矩阵，可以计算准确率、精确率、召回率和 F1-score 的值。

表 3 混淆矩阵

Table 3 Confusion matrix

	预测为正例	预测为负例
实际为正例	真正例(TP)	假正例(FP)
实际为负例	假负例(FN)	真负例(TN)

准确率：表示在所有样本中分对（即正样本被分为正，负样本被分为负）的样本数占总样本数的比例。

$$Accuracy=(TP+TN)/(TP+FP+TN+FN) \quad (12)$$

精确率：表示模型预测为正样本的样本中真正为正的比例。

$$Precision=TP/(TP+FP) \quad (13)$$

召回率：表示模型准确预测为正样本的数量占所有正样本数量的比例。

$$Recall=TP/(TP+FN) \quad (14)$$

F1-score：用来衡量二分类模型精确度的一种指标。它同时兼顾了分类模型的精确率和召回率。F1-score 可被看作模型精确率和召回率的一种加权平均。

$$F_1=2*\frac{precision*recall}{precision+recall} \quad (15)$$

5.3 实验结果与分析

5.3.1 参数调整

当设备接入网络时，网关记录该设备设置阶段的 n 个通信数据流量包以构建特征指纹。为了确定数据包数量 n 的取值，分别对数据包数量为 6，8，10，12，14，16 的情况进行了识别准确率的对比，实验结果如图 6 所示。

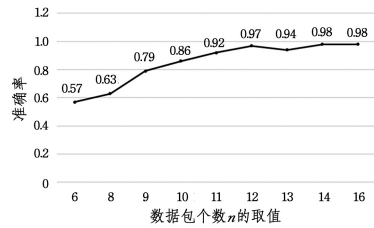

图 6 检测准确率随数据包数量的变化

Fig. 6 Detection accuracy varies with number of packets

图 6 中，横轴代表选取的数据包个数，纵轴代表检测准确率。可以看出，当数据包达到 12 个时，准确率达到 0.97，继续增加数据包个数后准确率趋于稳定，因此将最终特征指纹中的数据包个数确定为 12。经检测，此时设备识别训练和检测的平均耗时为 2.5 ms，满足实际需要。

5.3.2 结果评估

随机森林算法[10]被用来训练模型。在本文中，首先用随机森林做多分类器的设备识别模型 $C_{multi\_classifier}$，将数据集 D：$Set\{d_1,\cdots,d_n\}$ 随机分为训练集 $DS_{training}$ 和测试集 DS_{test}，多次实验得出的平均识别准确率为 67%，识别准确率较低。

为提高设备识别准确度，考虑为每一个设备型号训练二分类的设备识别模型 $C_{Double\_classifier}$，从指纹集 D：$Set\{d_1,\cdots,d_n\}$ 中选择设备型号 i 的指纹集 d_i，其余部分为其他设备的指纹集 d_x。对 d_i 与 d_x 二分类标记后，重新将其组合成用于二分类的数据集 D_{double}，再将 D_{double} 随机分成 5 份进行五倍交叉验证。实验得出每种设备识别的准确率，结果如图 7 所示。

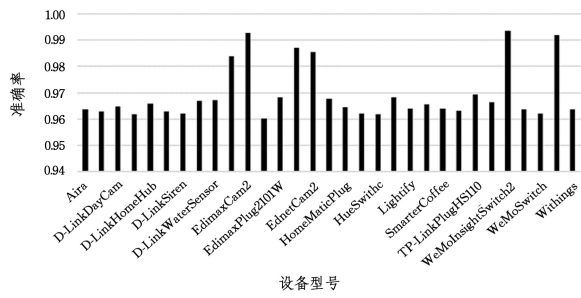

图 7 设备识别准确率

Fig. 7 Device identification accuracy

图 7 中，横轴代表 31 种设备类型，纵轴表示各个设备的识别准确率。可以看到，所有设备的设备识别准确率均达到 96%以上，某些设备甚至达到 98%以上。

为进一步评估实验效果，本文还计算了精确度、召回率以及 F1-Score 的值。实验结果如表 4 所列。

表 4 设备识别结果评估

Table 4 Device identification result evaluation

（单位：%）

设备类型	*Precision*	*Recall*	*F*1
Aria	95.65	98.3	97.2
HomeMaticPlug	94.58	92.3	93.41
Withings	96.2	97.8	97.1
MAXGateway	92.91	90.29	92.11
HueBridge	95.8	99.6	98.76
HueSwitch	94.62	98.15	95.77
EdnetGateway	96.78	99.23	98.96
EdnetCam1	78.64	69.9	75
EdnetCam2	75.74	66.35	69.19
EdimaxCam1	77.21	70.05	73.86
EdimaxCam2	68.49	73.93	71.07
Lightify	90.31	89.66	90.09
WeMoInsightSwitch1	70.84	73.25	72.22
WeMoInsightSwitch2	68.9	65.44	66.83
WeMoLink	95.57	97	96.3
WeMoswitch1	75.47	70.17	73.5
WeMoswitch2	77.26	78.39	77.91
D-LinkHomeHub	94.38	97.4	95.92
D-LinkDoorSensor	95.27	93.69	94.33
D-LinkDayCam	96.21	99.3	98.49
D-LinkCam	95.98	98.95	97.74
D-LinkSwitch	96.54	99.4	98
D-LinkWaterSensor	95.88	91.16	93.83
D-LinkSiren	96	97.2	96.43
D-LinkSensor	94.22	91.57	92.24
TP-LinkPlugHS110	96.51	99.88	98.35
TP-LinkPlugHS100	96.47	99.46	97.98
EdimaxPlug1101W	93.28	90.73	91.96
EdimaxPlug2101W	92.16	94.08	93
SmarterCoffee	96.1	98.71	97.05
iKettle2	93.92	96.5	94.19

可以看出，绝大多数设备的识别精度都比较高，但是对型号相似的设备的识别精确度比较低，比如 EdnetCam1 和 EdnetCam2。分析得出，这些设备生成的流量具有类似的特征，因此在分类时容易产生混淆。

5.3.3 方法对比

我们还将本文方法与文献[19]中的方法即特征调整前后的实验结果进行对比，结果如图 8 所示。

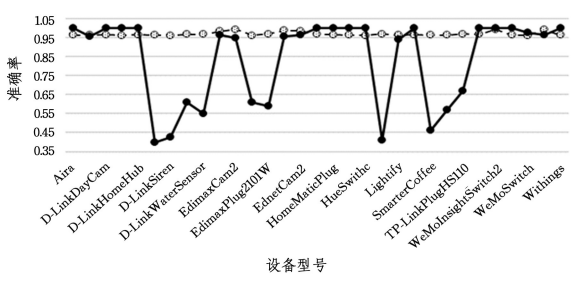

图 8 方法对比

Fig. 8 Method comparison

特征指纹调整后的结果中准确率普遍较高；而调整前即文献[19]方法的结果中，各个设备识别准确率之间的差异性比较大，其中有少数设备识的别准确率在 50%以下，但是也有很多的设备识别率达到 100%。相比而言，本文方法的稳定性更好。

准确率得到改善的原因在于对特征指纹的调整，比如去掉了不能有效体现差异性的 IP options 特征，然后将 IP 地址变换的频次作为指纹的一部分，将数据包长度特征转化为 10 以内的小数，避免指纹各像素点之间数据的差异过大而影响分类效果。特征构建指纹的方法会影响分类的准确率，在接下来的研究中我们会通过实验进一步分析比较不同的指纹构建过程对准确率的影响。

结束语 随着互联网的发展，特别是 5G 技术的成熟与应用，物联网必然迎来爆发式的发展，其安全问题成为了物联网得到广泛应用的关键，特别是复杂多样的物联网设备为大型组织内部网络带来了严峻的安全挑战。本文提出了基于流量特征提取和指纹构建的设备识别技术，对非白名单设备进行接入控制；介绍了一种基于设备识别和本体威胁建模的设备安全管理模型，用于对连接到内部网络中的设备实施安全管理。最后通过实验验证了基于随机森林算法的设备型号识别方法的识别准确率达到 96%以上，与类似方法相比有较好的稳定性，但是对相似设备的检测结果还不理想。

在未来的工作中，一方面，我们将改进所提模型，重点分析相似型号设备的流量特征选择；另一方面，对设备接入网络时的通信流量提取特征进行设备识别只是物联网安全研究中的一步，主要实现对非法设备的接入控制。我们计划在后续的工作中，再对设备在接入网络后、设备运行过程中的流量特征进行分析，主要实现对设备运行过程的安全威胁以及攻击行为进行实时的、全过程的检测，将过程监控与接入控制结合起来，实现对物联网设备网络的纵深防御。

参 考 文 献

[1] HOWELL J. Number of connected iot devices will surge to 125 billion by 2030. [EB/OL]. (2018-11-07)[2019-07-15]. https://technology.ihs.com/596542/.

[2] BORGIA E. The Internet of Things vision: Key features, applications and open issues[J]. Computer Communications, 2014, 1(1): 1-31.

[3] RESTUCCIA F, D'ORO S, MELODIA T. Securing the internet of things: New perspectives and research challenges[J]. IEEE Internet of Things Journal, 2018, 1(1): 1-14.

[4] STANKOVIC J A. Research directions for the internet of things [J]. IEEE Internet of Things Journal, 2014, 1(1): 3-9.

[5] PACHECO J, HARIRI S. IoT security framework for smart cyber infrastructures[C]// 2016 IEEE 1st International Workshops on Foundations and Applications of Self* Systems (FAS*W). IEEE, 2016: 242-247.

[6] CALERO. 3 Ways the Internet of Things will Impact Enterprise Security[EB/OL]. (2018-06-17)[2019-7-15]. https://www.calero.com/mobility-service-support/3-ways-the-internet-of-things-will-impact-enterprise-security/.

[7] BOZTAS A, RIETHOVEN A, ROELOFFS M. Smart TV foren-

sics: Digital traces on televisions. [EB/OL]. https://doi.org/10.1016/j.diin.2015.01.012.

[8] SAM B. WikiLeaks Dump Shows CIA Could Turn Smart TVs into Listening Devices [EB/OL]. https://theintercept.com/2017/03/07/wikileaks-dump-shows-cia-could-turn-smart-tvs-into-listening-devices.

[9] CACHE J. Fingerprinting 802.11 implementations via statistical analysis of the duration field[J]. Uninformed.org, 2006, 5.

[10] FRANKLIN J, MCCOY D, TABRIZ P, et al. Passive Data Link Layer 802.11 Wireless Device Driver Fingerprinting[C]// USENIX Security Symposium. 2006: 16-89.

[11] BOJINOV H, MICHALEVSKY Y, NAKIBLY G, et al. Mobile device identification via sensor fingerprinting[J]. arXiv: 1408.1416.

[12] VAN G T, SCHEEPERS W, PREUVENEERS D, et al. Accelerometer-based device fingerprinting for multi-factor mobile authentication[C]// International Symposium on Engineering Secure Software and Systems. Cham: Springer, 2016: 106-121.

[13] MEIDAN Y, BOHADANA M, SHABTAI A, et al. Detection of unauthorized iot devices using machine learning techniques[J]. arXiv: 1709.04647.

[14] NGUYEN T D, MARCHAL S, MIETTINEN M, et al. Diot: A crowdsourced self-learning approach for detecting compromised iot devices[J]. arXiv: 1804.07474.

[15] MEIDAN Y, BOHADANA M, SHABTAI A, et al. ProfilIoT: a machine learning approach for IoT device identification based on network traffic analysis[C]// Proceedings of the Symposium on Applied Computing. ACM, 2017: 506-509.

[16] SHAIKH F, BOU-HARB E, CRICHIGNO J, et al. A Machine Learning Model for Classifying Unsolicited IoT Devices by Observing Network Telescopes [C] // 2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC). IEEE, 2018: 938-943.

[17] SALMAN O, CHADDAD L, ELHAJJ I H, et al. Pushing intelligence to the network edge[C]// 2018 Fifth International Conference on Software Defined Systems (SDS). IEEE, 2018: 87-92.

[18] THANGAVELU V, DIVAKARAN D M, SAIRAM R, et al. Deft: A distributed iot fingerprinting technique[J]. IEEE Internet of Things Journal, 2018, 6(1): 940-952.

[19] MIETTINEN M, MARCHAL S, HAFEEZ I, et al. IoT Sentinel: Automated device-type identification for security enforcement in IoT[C]// 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS). IEEE, 2017: 2177-2184.

[20] DIRO A A, CHILAMKURTI N. Distributed attack detection scheme using deep learning approach for Internet of Things[J]. Future Generation Computer Systems, 2018, 82(1): 761-768.

[21] FAURI D, KAPSALAKIS M, DOSSANTOS D R, et al. Role Inference+ Anomaly Detection = Situational Awareness in BACnet Networks[C]// International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment. Cham: Springer, 2019: 461-481.

[22] MILOSLAVSKAYA N, TOLSTOY A. Internet of Things: information security challenges and solutions[J]. Cluster Computing, 2019, 1(1): 1-17.

[23] NAWIR M, AMIR A, YAAKOB N, et al. Internet of Things (IoT): Taxonomy of security attacks[C]// 2016 3rd International Conference on Electronic Design (ICED). IEEE, 2016: 321-326.

[24] PACHECO J, ZHU X, BADR Y, et al. Enabling risk management for smart infrastructures with an anomaly behavior analysis intrusion detection system[C]// 2017 IEEE 2nd International Workshops on Foundations and Applications of Self* Systems (FAS* W). IEEE, 2017: 324-328.

[25] MOZZAQUATRO B, AGOSTINHO C, GONCALVES D, et al. An Ontology-Based Cybersecurity Framework for the Internet of Things[J]. Sensors, 2018, 18(9): 3053-3061.

[26] MOZZAQUATRO B A, JARDIM-GONCALVES R, Agostinho C. Towards a Reference Ontology for Security in the Internet of Things[C]// IEEE International Workshop on Measurement & Networking 2015. IEEE, 2015: 289-296.

[27] HERZOG A, SHAHMEHRI N, DUMA C. An ontology of information security[J]. International Journal of Information Security and Privacy (IJISP), 2007, 1(4): 1-23.

[28] FENZ S, EKELHART A. Formalizing information security knowledge[C]// Proceedings of the 4th International Symposium on Information, Computer, and Communications Security. ACM, 2009: 183-194.

[29] UNDERCOFFER J, JOSHI A, PINKSTON J. Modeling computer attacks: An ontology for intrusion detection[C]// International Workshop on Recent Advances in Intrusion Detection. Berlin: Springer, 2003: 113-135.

[30] ACRIS. IoT devices setup captures (IoT Sentinel experiments) [EB/OL]. https:// research. Aalto. fi/files/1150458/captures IoT Sentinel. zip.

YANG Wei-chao, born in 1991, M. S. candidate. His research interests include security of internet of things and so on.

GUO Yuan-bo, born in 1975, Ph.D, professor. His research interests include network attack and defense and so on.