基于搜索的物联网设备识别框架

邹宇驰[1,2], 刘 松[1,2], 于 楠[2], 朱红松[1,2], 孙利民[1,2], 李 红[1,2], 王 旭[1,2]

[1] 中国科学院大学网络空间安全学院 北京中国 100049
[2] 中国科学院信息工程研究所物联网信息安全技术北京市重点实验室 北京中国 100093

摘要 越来越多的物联网设备接入到互联网中, 但由于设计上的缺陷或者缺乏安全防护手段, 这些暴露在公网上的物联网设备极容易受到黑客的攻击与利用。研究表明, 具有相似产品属性的物联网设备很有可能存在相同漏洞, 因此有效的识别网络空间中的物联网设备, 对其产品属性, 如设备品牌、型号等相关信息进行细粒度识别和标定, 对把握网络空间实体设备的安全态势具有重要意义。本文提出一种基于搜索的物联网设备识别框架, 利用物联网设备协议标语中富含的产品属性信息, 通过自动化网络搜索技术构建物联网设备信息库, 进而实现对未知新设备细粒度地自动分级识别和标定。通过公网实验, 该框架能够很好识别视频监控和工控设备的产品属性, 型号识别准确率均超过 90%。

关键词 物联网安全; 设备产品属性识别; 协议标语; 细粒度
中图法分类号 TN915.08 **DOI 号** 10.19363/J.cnki.cn10-1380/tn.2018.07.03

IoT Device Recognition Framework based on Web search

ZOU Yuchi[1,2], LIU Song[1,2], YU Nan[2], ZHU Hongsong[1,2], SUN Limin[1,2], LI Hong[1,2], WANG Xu[1,2]

[1] School of CyberSpace Security, University of Chinese Academy of Sciences, Beijing 100049, China
[2] Beijing Key Laboratory of IOT information security, Institute of Information Engineering, CAS, Beijing, 100093, China

Abstract More and more internet of things (IoT) devices are connected to the Internet, but many of them have design defects and less security consideration because of lower price and limit resources. These devices, therefore, are more easily cracked by malicious attackers by utilizing various implementation vulnerabilities. It is well known that IoT devices from same manufactures were tend to have same vulnerabilities, so we can obtain some valuable hints through the brand and model of devices without the need to verified the vulnerabilities one by one while evaluating devices' security status. Some research proposed methods to identify the categories or manufactures of IoT devices, but the information is so coarse that many devices may be marked with wrong security tags. In this paper, we proposed a IoT devices recognition framework based on Web search, which identified IoT devices in fine-grained manner, the brand and model, by matching their protocol banners with the products attributes database collected from specific electronic business Webs. Because the collecting is never end, we can recognize more and more IoT devices as some new products are found and put into the database. Internet experiments showed that, with our framework, the recognition accuracy on brands and models, for video surveillance and industrial control equipment, exceeds 90%.

Key words internet of things security; products attributes recognition; protocol banners; fine-grained manner

1 引言

随着物联网(Internet of Things, IoT)技术的飞速发展, 各种类别、类型、品牌、型号的物联网设备在日常生活中发挥着重要的作用, 如家用路由器、IP Camera、网络打印机、工业物联网中的工业控制系统(Industrial Control System)等。有报告显示[1], 目前有超过 50 亿的物联网设备, 这个数量在 2020 年将会达到 200 亿。这些设备在被分配公网 IP 地址的情况下, 可以借助互联网较为方便地直接与之交互或管理。但随之而来的是这些设备缺乏安全防护或者设计上的缺陷或者软件漏洞的曝光以及黑客利用这些不安全因素实施恶意行为。如 2016 年 10 月份美国东海岸断网事件, 归咎为大量的 IP Camera 存在弱密钥的缺陷。由此可见, 物联网设备给网络空间带来的安全问题不容忽视。

通讯作者: 于楠, 硕士, 助理研究员, Email: yunan@iie.ac.cn。
本课题得到国家重点研发计划(No.2016YFB0801303-1); 自然基金面上项目(No.U1536107)和中国科学院信息工程研究所国际合作项目(No.Y7Z0451104)资助。
收稿日期: 2018-03-30; 修改日期: 2018-05-30; 定稿日期: 2018-06-19

经过调研发现, 相同品牌或相同型号的设备会存在相同漏洞, 如 CVE-2015-7254[2], 影响了华为路由器下的 HG532e、HG532n、HG532s 等三种型号。因此, 在网络空间中快速、准确地识别出物联网设备, 细粒度地判断其产品属性, 再通过漏洞库进行准确标识, 既能有助于建立物联网设备安全分布态势图, 又能帮助管理员加固设备防护, 加强资产管理, 帮助后续制定防护策略, 为安全防护方案提供参考。Shodan[3]与 Censys[4]是目前商用化最好的面向实体设备的搜索服务系统, 是开展安全研究的重要资源平台。

本文提出一种基于搜索的物联网设备产品属性识别框架, 着重提升对物联网设备产品相关属性的识别能力。通过实时地对类型、品牌、型号库搜索更新, 实现对不同类型、不同厂商甚至不同型号的物联网设备的识别能力的提升。本文通过属性信息库自动化构建框架、数据获取与预处理模块、设备产品属性分级识别模块三个主要部分定义了物联网设备产品属性完整识别框架。

首先, 通过爬虫等自动化的抓取手段, 自动化地、实时地搜索电商平台上出现的物联网设备产品属性相关信息, 不断更新设备信息库。其次, 建立全连接之后, 发送针对特定协议或端口的特定探测报文, 获得物联网设备返回的协议标语信息, 利用自然语言处理(Natural Language Process, NLP)的方式, 除去协议标语中的停用词、特殊符号等非关键性因素后, 再对标语信息进行分词。最后利用自动化收集的信息库以及相关内容过滤标语信息中的标识信息, 如类型、品牌、型号等。

通过对以上研究方案的实现, 克服了对物联网设备识别能力不全、识别粒度较粗的问题。目前可识别品牌种类库达到 1200 种以上, 型号种类库达到 12000 种以上, 构建效率远高于人工指纹模式, 且自动形成对新品类设备的识别能力。通过在真实的公网环境中的实测,本框架利用 Onvif 协议标语对公网视频监控设备型号识别准确率最高达到 97%, 利用 FTP 协议标语对型号识别准确率达到 91%, 利用 Ethernet/ip 协议标语对工业控制系统设备型号识别准确率达到 97%, 利用 Bacnet 协议标语对工业控制系统型号识别准确率可达到 98%。

2 相关工作

近些年来对于物联网设备的识别研究工作逐渐成为一个热点。2005 年, Kohno 等[5]根据设备硬件中存在的微小偏差, 利用时钟偏移值, 实现对远程设备的指纹识别技术; 2010 年, Cui 等[6]对暴露在公网上的弱口令设备进行类别以及类型的分析, 可识别范围从企业设备(防火墙、路由器)到消费电子设备(VoIP, IPTV 机顶盒)等; 2015 年, Radhakrishnan 等[7]提出基于人工神经网络的指纹识别算法 GTID 来进行物联网设备类型识别, 能达到较好的精确率; 2016 年,曹来成等[8,9]通过提取首页 http 数据包头部字段和状态码作为设备特征, 基于设备特征向量之间的余弦相似度, 通过 K-means 聚类方法实现对设备的划分; 同年还提出了网络空间终端识别框架, 该框架利用 http 的标语信息和 HTML 源代码双重因素进行终端设备品牌识别; 2017 年, 任春林等[10] 通过机器学习的方法, 能够根据 WEB 页面信息识别设备是否是视频监控设备; 同年, Miettinen 等[11]提出对特定网络中物联网设备类型的识别方法; Li 等[12,13]通过视频监控 WEB 页面提出一种自动化的视频监控设备分类方法; 同年, 他们通过设备登录页面的特征, 提出了一种 GUIDE 的设备识别框架, 该框架首先通过特征提取方法筛选出页面的关键字特征, 然后通过构建分类器进行视频监控设备的识别, 达到了较高的识别正确率; Meidan 等[14]应用机器学习算法于网络流量数据, 从而准确识别连接到网络的物联网设备类型; 2018 年, Bezawada 等[15]提出了基于物联网设备指纹的类型识别方法, 通过从网络流量中提取设备行为的近似特征, 用于训练设备类型的机器学习模型; Shaikh 等[16]提出一种机器学习的方法实现对网络空间中物联网设备的二分类模型, 实现对网络空间中恶意的物联网设备活动的准确识别。用于搜索 SCADA 工控设备的 Modscan[17]工具和用于发现西门子 PLC 设备的 Plcscan[18]工具则是利用专有协议的标语信息进行设备识别,通过人工构建对应的设备专有协议指纹库, 对设备的专有协议标语信息进行识别, 从而对设备进行识别分类, 区分出设备的类型; Feng 等[19]分析了 17 个常用工控专有协议, 提出了针对互联网工控系统设备的识别指纹。

但是, 以上研究对物联网设备的识别工作仍有些许不足。人工提取指纹能帮助解决一部分识别, 但是人工指纹容易退化, 对较新的、相似型号的设备识别能力不够, 识别效率较低。引入机器学习或者深度学习的方法对网络流量进行分析, 虽然提高了设备识别的自动化程度, 但是当前只能识别到设备类型或设备品牌, 识别粒度较粗。而相同品牌的不同型号设备的页面、标语甚至流量特征信息很类似, 无法良好区分出设备的型号。对于新出现的设备, 已有的设备识别分类器将会失效, 需要重新训练分类器, 而

且重新训练的代价较高, 并且对于不同类型设备可能提取的特征会有所不同。

经过调研后发现, 与物联网设备进行有效连接后, 物联网设备返回的报文头部常常带有丰富的与产品属性相关的信息, 如设备类型、品牌、型号等, 这部分信息不仅仅反映了设备基本情况, 同时还包含了丰富的语义信息, 如已知型号可以帮助推断设备类型与品牌, 已知类型与型号可以推断品牌。属性信息之间的互相关联能增加对设备的判别能力, 提高识别准确率。

根据物联网设备产品属性特性, 通过搜索的方式, 自动化构建了产品属性信息库, 实现了物联网设备识别框架。通过信息库的建立, 可以准确且高效的针对不同协议标语情况, 提出相对应的解析、识别方法, 更能获得未曾在标语中出现的产品属性, 从而提升对网络空间中整体物联网设备的识别能力。

本文围绕物联网安全状态分析的现实需求, 以实现准确、高效、细粒度地物联网设备在线识别为目标, 在现有设备网络属性识别方法的基础上, 研究物联网设备产品属性的网络识别框架, 为进一步研究网络空间中物联网系统安全问题打下坚实的基础。

3 物联网设备识别框架

3.1 概述

为了实现对物联网设备准确、细粒度的识别, 本文设计了如图 1 所示物联网设备识别框架, 总共分为三部分: 属性信息库自动化构建框架, 数据采集与预处理模块, 设备产品属性分级识别模块。属性信息库自动化构建框架自动地从互联网电商平台搜索产品属性等相关信息, 构建产品属性信息库。数据采集和预处理模块获取标语信息模块, 并剔除不相关内容。设备产品属性分级识别模块对处理后的标语信息进行分级提取, 分别提取出设备类型、品牌、型号等, 再根据标语信息中出现的语义信息推断设备基本信息。

3.2 属性信息库自动化构建框架

产品属性库自动化构建是识别物联网设备产品属性的基础, 同时也是发现新物联网设备品牌、型号的有效手段。目前网络空间中物联网设备种类纷繁复杂, 相同类型的相同品牌有多种型号, 若都采用人工收集的方式无疑大大增加了工作难度, 还可能导致识别滞后的问题。如何有效地收集、统计物联网设备的类别、类型、品牌、型号等关键属性, 是本文需要解决的问题之一。因此, 本文采用自动化地方式构建产品属性信息库, 可以有效降低人工参与品牌型号收集工作, 提升收集效率。

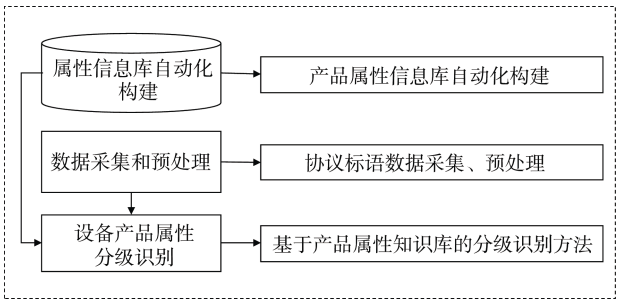

图 1 物联网设备识别框架

Figure 1 IoT device recognition framework

3.2.1 物联网产品属性分类

网络空间中的物联网设备纷杂多样, 种类繁多。无论是用于安防的视频监控系统还是用于基础设施的供气供电的工业控制系统, 其中涵盖的物联网设备也是多种多样, 设备之间关系复杂。如图 2 展示的是安防系统、家居系统和智能电网系统。从图中可以得知, 安防系统中既有视频监控设备 IP Camera(网络摄像机)、NVR(Network Video Recorder, 网络硬盘录像机), 又有交换机和网关; 家居系统中既有智能家居设备智能开关、智能吊灯, 又有路由器设备; 智能电网系统中既有工控设备 PLC(Programmable Logic Controller, 可编程逻辑控制器)、SCADA (Supervisory Control And Data Acquisition,数据采集与监视控制设备), 同样也有交换机设备。难点在于这三个系统中的设备既有其独特性又有重复, 若没有一个合理的设备的属性分层标准, 则很难将物联网设备的产品属性划分清楚。

对于上文的描述, 提出一种属性分层方式, 即各系统中都可能出现的设备划分为一层, 每个系统中独特的设备划分到一层。这样可以有效的解决设备类型分类的问题, 但如何确定每一类具体的内容以及与品牌型号的关系, 则需要确定一个标准。

物联网设备种类丰富, 同一种类的物联网设备由很多厂商生产, 并且同一品牌同一类型的物联网设备有多种不同的型号。Hikvision(海康威视)的 IP Camera 的类型有很多, 如 DS-2DF1-611、DS-2CD1-203 代表的是两个不同型号的海康威视的视频监控设备。对于物联网设备的所属类别来说, 若直接把设备都归类为物联网设备, 则其划分太过于粗糙, 不能很好地确定此设备的具体用途, 因此需要在设备类型前再设置一个类别, 来划分物联网设备所属的具体的种类。

图 2　纷繁多样的物联网设备

Figure 2　Diverse IoT devices

因此, 可以将设备的产品属性以层级关系定义。定义的产品属性有设备类别、设备类型、设备品牌和设备型号。其特性如下:

• 设备类别: 指的是设备归属的系统, 表示的含义是设备的主要用途。比如视频监控系统、工业控制系统和路由交换系统等。

• 设备类型: 指的是具有相同本质特点的同类设备, 表示的含义是设备的名称。比如视频监控系统下的 NVR, 工业控制系统下的 PLC。

• 设备品牌: 指的是设备的品牌, 表示设备的所属。比如海康威视、Simens(西门子)、Scneider(施耐德)。

• 设备型号: 指的是设备的具体型号。比如 DS-2DF1-611 代表的是海康威视的 IP Camera, S7-200 代表的是西门子的 PLC。

通过确定需要识别的设备产品属性的定义标准, 则可以得知设备类别、设备类型、设备品牌和设备型号的关系, 划分标准如图 3 所示。通过这种划分方式, 发现物联网设备的产品属性之间具有交叉和继承关系, 交叉关系是指同一类别的设备如 DVR 可以被不同厂商生产, 如 Dahua 或者海康威视; 继承关系是指已知某个设备是 DVR, 那么可以推断其肯定是视频监控设备。通过对以上属性的分析, 本文构建了 10 种类别库, 53 种类型库, 部分结果如表 1 所示。

表 1　设备类别分类表

Table 1　IoT device category classification table

英文简称	英文全称	中文全称	描述
ICS	Industrial Contr-ol System	工业控制系统	主要指工业控制的 DCS/SCADA/ PLC 等设备
VSS	Video Surveilla-nce System	视频监控系统	主要指视频监控系统和视频监控设备等

3.2.2　设备品牌库构建

构建品牌库的目的是通过品牌库中的品牌特性与待识别设备的协议标语信息进行比较, 过滤并识别出待识别设备的标语信息中的品牌。同时得到设备的厂商, 品牌参数, 设备品牌的相关描述以及所属类别等信息。

通过对设备的协议标语信息观察发现, 设备的协议标语信息中出现的设备品牌信息基本上都是品牌英文名称。图 4 展示的是 FTP、TELNET 等协议标语信息, 如 FTP 协议标语中出现品牌名 MikroTik。因此有必要对设备品牌建立一个完善的库, 不仅只存储品牌名, 还需存储与之相关的信息, 如品牌描述、品牌所属国家等。这些信息有助于更好地对品牌属性进行刻画。因此设计如表 2 所示的品牌库结构。设备的所属国家字段, 其目的是为了更好的统计品牌库中收集的品牌数量在国内外的分布情况。对于品牌和设备厂商的关系, 一个设备厂商下会有多个不同的品牌, 但是一个品牌只能属于一个设备厂商。而对于在设备品牌库中设置设备品牌链接和设备品牌描述信息字段, 则是为了方便进一步了解设备品牌的其它具体信息, 更好和更全面的掌握识别出来品牌设备的其他产品属性。

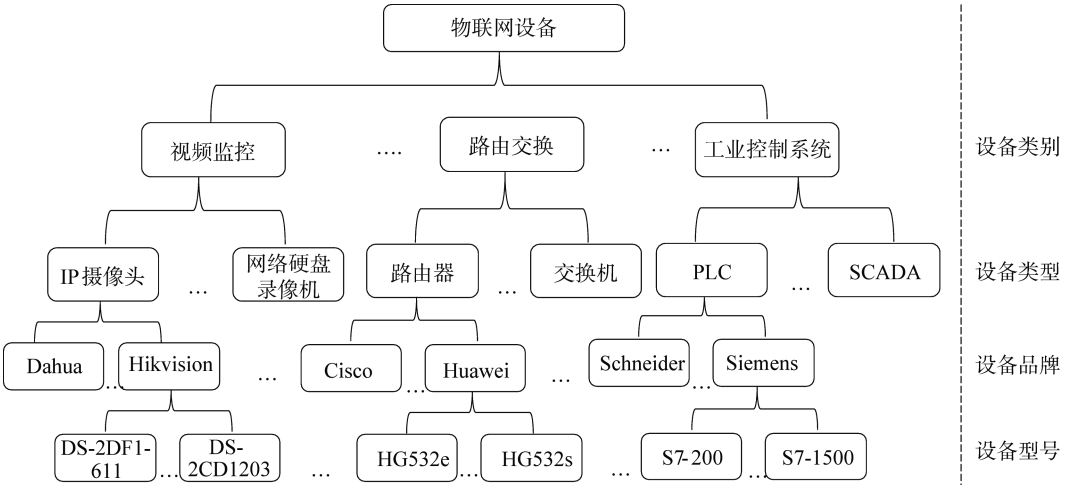

图 3　物联网设备属性划分标准

Figure 3　Standard for attribute classification of IoT device

表 2　设备品牌库结构

Table 2　Structure of IoT device manufacture database

字段名称	字段描述	字段说明
id	品牌 id	此 id 为自增 id, 方便统计品牌数量
en_name	品牌英文名	主要为品牌的官方名称, 也是协议标语信息中主要出现的品牌名称
cn_name	品牌中文名	品牌对应的中文名称
country	国家	品牌所属的国家
description	品牌描述	品牌的相关描述信息
brand_link	品牌官网链接	品牌的官方网站链接
manufacture	品牌厂商	品牌所属的设备厂商
update_time	更新时间	记录品牌库中更新时间

表 3　设备型号库结构

Table 3　Structure of IoT device model database

字段名称	字段描述	字段说明
id	型号 id	设备型号 id
model	型号名	具体的型号
device_brand	品牌名	品牌英文名
device_type	类型名	类型英文名缩写
device_category	类别名	类别英文名缩写
description	描述	型号的描述信息
model_link	型号链接	某型号的设备的详细介绍链接
model_picture	型号的图片	某型号的设备的图片
update_time	更新时间	型号信息的最近一次更新时间

3.2.3　设备型号库构建

设备型号库构建的目的是通过型号库中的设备型号信息与待识别的协议标语信息比较, 过滤出待识别设备标语信息中的型号信息, 从而标记出设备的型号, 根据设备产品属性信息库的划分标准, 进一步推导出设备所属类型、类别以及品牌信息。通过构建的设备型号库可知设备的品牌信息, 通过得到的设备品牌信息, 则可以在品牌库中进一步查询获取设备品牌相关的其他信息。如图 4 所示, 图中 TEL-NET 协议中的 BCM96338 是设备型号, Router 代表其设备类型, 根据型号库, 可以推断其品牌为 Beet-el, 同样已知了设备的型号信息之后, 可以获取设备类别类型信息。因此, 构建表 3 所示的型号库结构, 增加该型号对应的品牌、类型、类别信息。

3.2.4　设备品牌型号库自动化收集框架

在品牌型号库模型构建好后, 下一步则为收集设备品牌和型号等属性信息。但是由于物联网设备海量异构, 物联网设备品牌和型号种类庞大, 若靠人工收集物联网设备品牌和型号信息, 难度较大。一是人工收集成本太高, 二是人工收集对于发现新设备品牌和型号也相对滞后。

一般情况下, 设备品牌和型号信息可以在各厂商的官网上查到对应的信息, 常规的品牌型号爬取需要对各厂商的网站分别进行爬取, 过程较为烦琐。但发现各大物联网设备厂商下属有多个代理厂商, 代理厂商会将设备基本属性公布在第三方电商平台网站上, 如亚马逊, ZOL, IT168 等。图 5 显示的是 ZOL 上搜索打印机显示结果, 可以发现产品属性品牌和型号都结构化的呈现在网页上, 包括该设备是否具备联网功能, 对于不具备联网功能的型号, 收集框架并不会将其爬取, 如爱普生 R330 这款打印机, 收集框架并不会将其收录。通过对这部分第三方电商平台上的设备基本属性信息进行爬取与收集, 可以降低构造爬虫的开销, 同时收集更为方便。

本文依据此类网站网页展示的品牌、型号等结构化特性, 提出一种设备品牌、型号自动化收集框架。自动化的爬取和收集设备品牌、型号等属性, 使

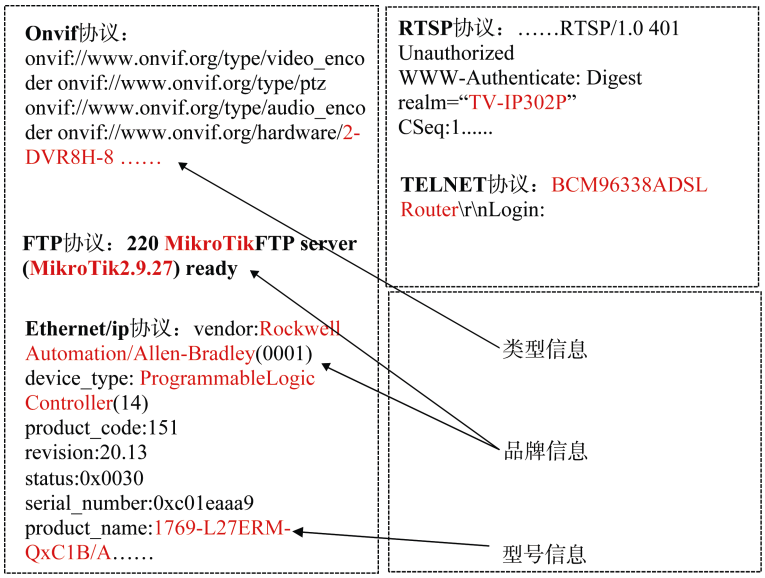

图 4　协议标语部分信息

Figure 4　Part information of protocol banners

得收集设备品牌和型号的方法更具有一般性，若有新的需求可以增加对应的爬虫，具有良好扩展性，并能周期性更新和收集品牌和型号。

图 5　电商平台结构化的品牌型号

Figure 5　Structured manufacture and model of E-commerce platform

图 6 为品牌型号库自动化收集框架，基于 Python 的 Scrapy 架构，使用异步网络框架 Twisted，实现了对产品属性的自动化收集工作。框架分为四个过程：启动过程，数据解析过程，数据获取归一化过程，自动入库过程。

(1) 启动过程

根据需求，设定了两套启动方案，其一为主动启动，方便配置待爬取网站，待爬取类型、品牌、型号等属性信息。其二为定时启动，通过设定爬取周期，定时对特定网站进行爬取。

(2) 数据解析过程

数据解析过程包含两个步骤，首先启动爬虫框架对目标网页进行特定信息收集，即对设备产品属性页面进行特定信息收集；然后使用对应网页的解析规则获取产品属性元素，即品牌、型号以及基本参数。对于非联网设备进行剔除，因为本框架对物联网的识别是基于远程连接获取的标语信息，对于非联网的设备不适用。

根据不同网站的页面结构及其特性，需要针对对应网站建立产品属性提取规则，构建对应网站页面解析器，完成对产品属性的解析。在提取设备页面上的品牌型号信息时，需要提取设备的品牌、设备的型号、型号的相关描述信息、型号的标签信息以及设备型号图片等。

(3) 数据获取归一化过程

数据获取归一化，其目的是将(2)过程得到的页面面元素信息归一化为品牌元素(品牌中文、品牌英文和品牌参数)，型号元素(型号名称、型号标签、对应型号设备链接、型号详情、型号图片和型号参数)和类型元素。其中品牌元素的中英文通过程序调用有道词典 API 自动获取。

(4) 自动化入库过程

根据数据获取归一化过程可以得到品牌、型号和类型的三元组关系。将品牌元素自动存储到品牌库中，将品牌英文，型号元素，类型元素以及根据产品属性分层标准得到的类别信息自动存储到型号库中。从而完成了品牌库和型号库数据自动收集过程。

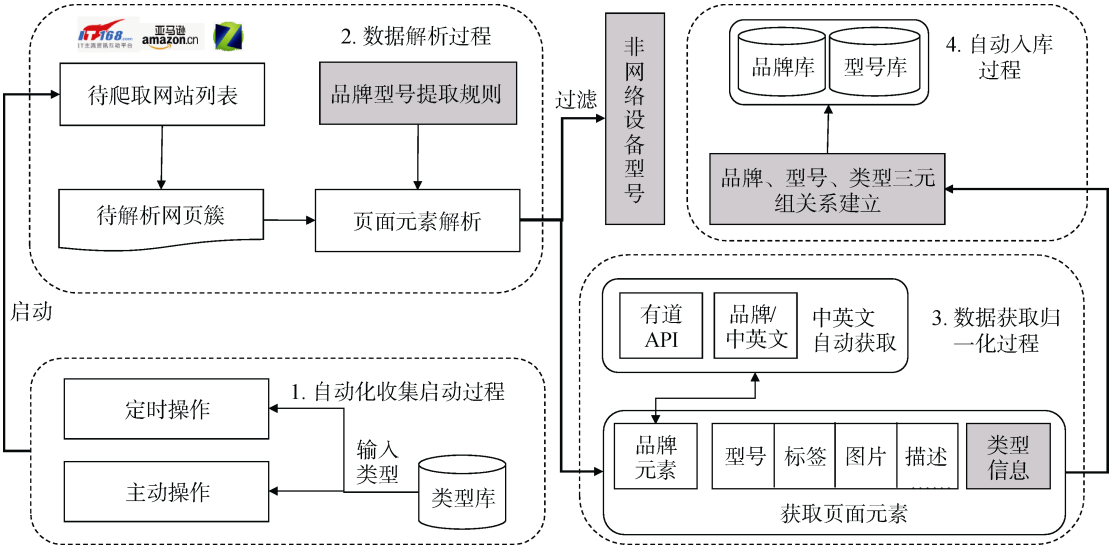

图 6　物联网设备品牌型号自动化收集框架

Figure 6　Automatic collecting framework of IoT device manufacture and model

如图 7 所示为自动化收集框架从 2017 年 9 月份至 2018 年 1 月份累计收集到的品牌与型号数量情况，可以发现，到 2018 年 1 月份，去重后的品牌种类达到 2397 种，去重后的型号种类达到 56282。

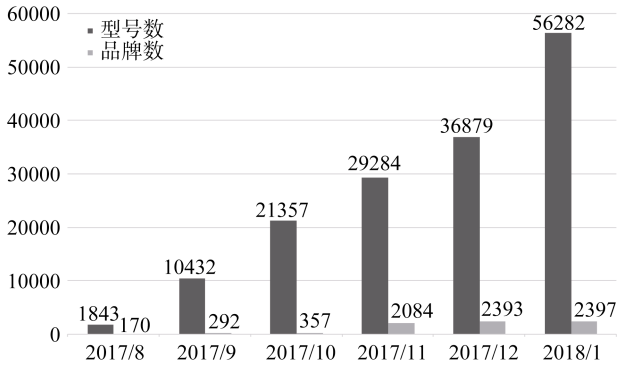

图 7　设备品牌型号统计图

Figure 7　IoT device manufacture and model statistics

2017 年 8 月份代表当时收集到的人工指纹情况。到 2018 年 1 月份，与人工指纹相比，自动化收集框架新增品牌种类 2227 种，新增型号种类 54439 种。图 8 展示的为随着时间的推移，工业控制系统、路由交换系统以及视频监控系统三类不同设备类别识别数量的变化情况。对图 8 的分析发现，从 2017 年 9 月份到 2017 年 12 月份，视频监控系统识别数量保持持续性增长，但从 2018 年开始出现下跌，同样变化的还有工控系统。原因在于之前被识别为视频监控系统的设备出现了一定的误报率，经过对算法的改进，将识别准确率提升到 90%以上，因而剔除了这些误报设备,导致了识别数量的下降。同样情况的还有工控系统。而可识别的路由交换系统的数量，在保证较高准确率的前提下，实现了与品牌型号数量的同步增长。

3.3　标语数据采集与预处理模块

标语采集是指采集全网或者特定网络范围内的基于 TCP 或者 UDP 协议的应用层协议标语信息。利用全连接或半连接的方式发现设备存活端口，接着对这些设备存活端口构造并发送特定协议探测包，获取存活端口的响应信息，这部分响应包信息被称为协议标语信息。图 9 显示的是一种基于 TCP 协议获取协议标语信息的方式。

协议标语信息分为两种，一种是可读字符串标语，一种是不可读字符串标语。针对不同协议标语设计不同的预处理流程。

对可读协议标语，首先需要过滤掉非物联网设备，为了过滤此类非物联网设备的协议标语信息，建立了一个非物联网设备关键词库，如 Nginx 或者 Apache 等，这种词汇标识了该设备有很大概率为 WEB 服务器。判读特定协议标语在特定字段上的词是否在非物联网设备关键词库内，若在则标记此设备为非物联网设备，排除掉此设备。接着需要把冗余部分信息剔除，如 Telnet 协议标语信息，其中存在不少冗余信息，如\r\n 等，需要将特殊符号、标点符号以及不可打印字符剔除。对于单个词语中的词语长度小于 3 的，直接删除该词语；然后根据过滤规则库中的规则，进一步过滤掉混淆字符，比如日期、无用数字等。为了进一步减少不相关字符串的干扰，对剩余的协议标语信息使用自然语言处理的方式，将 NLTK(Natural Language Toolkit)库中收集的英文停用词在协议标语中删除。接着使用 NLTK 库中

Tokenize 分词工具对剩余协议标语进行分词, 得到分词后的标语词汇列表。

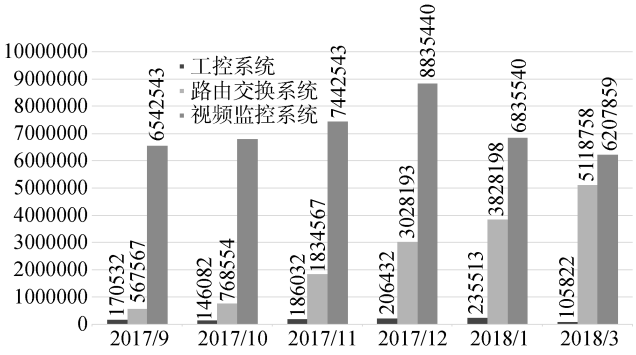

图 8 类别识别数量统计图

Figure 8 Recognizable category statistics

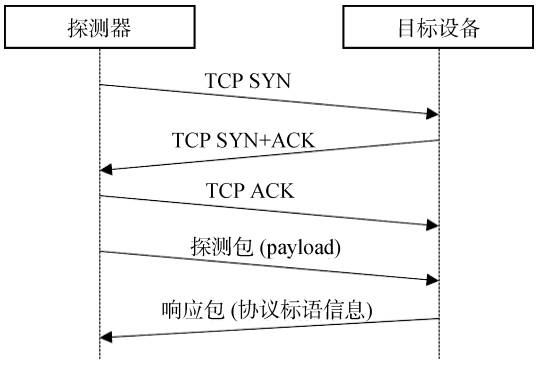

图 9 TCP 协议标语抓取

Figure 9 Capturing TCP banner

对于不可读字符串协议, 需要将其转码后, 得到可读的形式, 然后复用可读字符串处理流程。

经过以上预处理后的标语信息, 是后续用于设备产品属性识别的关键基础。

3.4 设备产品属性分级识别模块

图 10 显示了 FTP、RTSP 等 8 种不同类型的协议标语信息, 根据协议的标语信息分析可以发现, 若使用产品属性信息库与协议标语直接比较存在许多困难, 其困难主要体现在: (1)协议种类多样化, 有的通用协议如 SNMP 返回 16 进制字符串, 有的通用协议如 Http 返回字符串, 还有一些工控设备专有协议如 Ethernet/ip 返回是键值对的形式。协议标语内容格式不一致直接导致识别方法无法提取有效的信息来进行识别, 需要确定一种让识别更加有效的数据提取方案。(2)协议标语信息内容复杂, 在设备的协议标语信息中, 设备信息杂糅多样, 有的标语信息中只有类型、品牌或者型号信息的一种, 有的标语信息中含有此三种属性的两种或者三种。

为了解决以上困难, 在保证型号识别准确率和品牌型号召回率的情况下, 本论文提出并实现基于

FTP协议:220 MikroTikFTP server (MikroTik6.28) ready\r\n

RTSP协议:RTSP/1.0 401 Unauthorized\r\nCSeq: 2\r\nWWW-Authenticate: Digest realm=\"Hikvision\", nonce=\"18b81359a12c32308156df26d7da69ee\", stale=\"FALSE\"\r\nWWW-Authenticate: Basic realm=\"^\"\r\n\r\n

SNMP协议:303e020101040670756 26c6963a2310205ffdc63c29a020100020100302230200 6082b0601020101010004145 26f757465724f5320434352313031362d313247

Telnet协议:BCM96338 ADSL Router\r\nLogin:

Onvif协议:onvif://www.onvif.org/type/video_encoderonvif://www.onvif.org/hardware/2-DVR8H-8

Ethernet/ip协议:….vendor: Rockwell Automation/Allen-Bradley(0001)\r\n

device_type: Programmable Logic Controller(14)

product_code:151\r\n revision:20.13 \r\n status:0x0030 \r\n

serial_number:0xc01eaaa9\r\nproduct_name:1769-L27ERM-QxC1B/A

Http协议:\\u003cli\\u003e\\u003ca href=\"https://www.startech.com.bd/Security-Camera/dvr-nvr-camera-security-solutions/dahua-nvr\"\\u003eDahua NVR\\u003c/

图 10 多种类型协议标语

Figure 10 A great diversity of protocol banners

搜索的设备产品属性识别框架。如图 11 所示, 本框架分为四个部分, 分别为数据采集部分, 数据预处理部分, 信息库匹配部分和人工验证反馈部分。

针对困难一实现了特定协议解析器方法, 针对特定协议标语内容格式, 如 3.3 节所述的方案采用对应协议解析器, 完成协议有效信息的获取与预处理; 本框架针对困难二实现了分级识别方法, 按照顺序依次识别出协议信息中的类型、品牌和型号。由于 3.3 节介绍了相关数据采集与预处理部分, 因此本节着重介绍属性信息库匹配和人工验证反馈部分。

选择类型、品牌、型号过滤顺序的原因是, 协议标语的语义信息对于识别研究是非常重要的, 比如三星有款打印机型号名为 100, 若直接用型号名匹配, 对于返回标语信息中带有 100 字样的都会被识别为三星打印机, 这种完全脱离语义的识别模式大大降低识别准确率。但是, 当配合上类型或者品牌信息, 可以有效对 100 这个字符串进行筛选, 也就是说没有过滤品牌或者类型, 单纯只有型号, 该设备并不会被成功识别。

因此, 本文采用类型或者品牌结合型号的方式共同对设备进行识别。其次, 利用信息库可以获取除设备类型、品牌、型号之外的其他设备产品属性信息, 比如设备厂商、官网等。图 12 显示信息库匹配的 3 个阶段: (1)类型匹配阶段; (2)品牌匹配阶段; (3)型号匹配阶段。通过以上 3 个阶段, 可以过滤大部分物联网设备产品基本信息。最后本文通过验证反馈的方式, 进一步提高设备识别准确率。

3.4.1 设备类型匹配阶段

物联网设备类型匹配阶段, 是使用类型库中的类型与标语词汇列表中的词进行匹配, 得到设备的类型, 如算法 1 所示。因为标语与类型集合中存在大小写, 而匹配算法对大小写敏感, 因而需要将标语词与类型集合全转为小写, 接着把标语中分词后的列表中 L_{banner} 每个元素与已有设备类型集合 S_{type} 进行比较, 最终得到过滤类型后的标语词汇列表(L_{ftbl})。

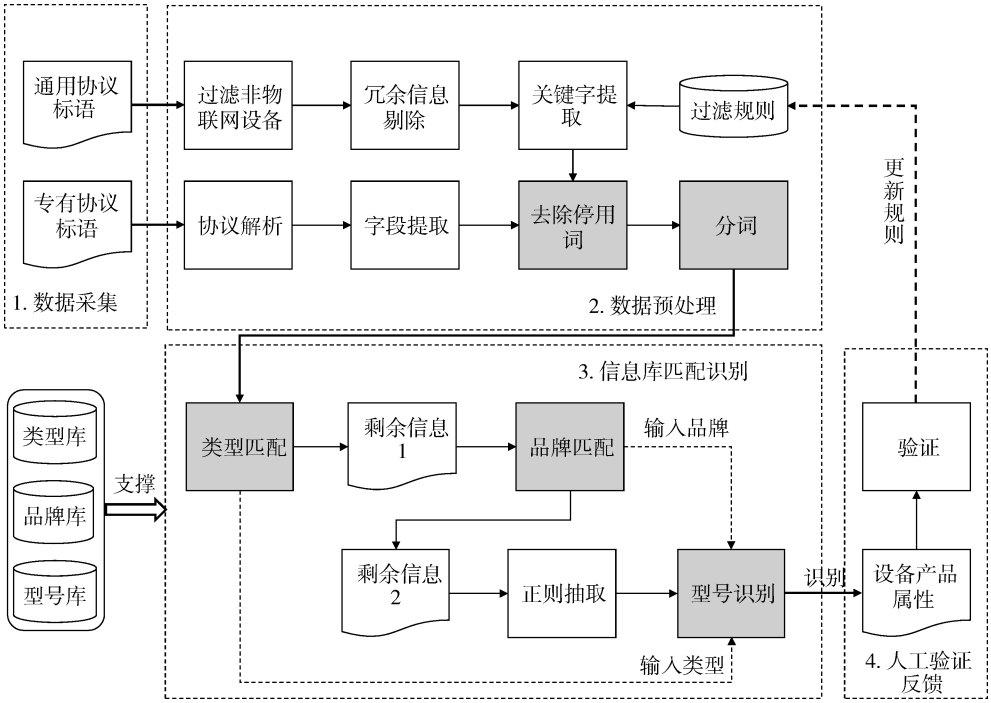

图 11　产品属性识别框架

Figure 11　Product properties recognition framework

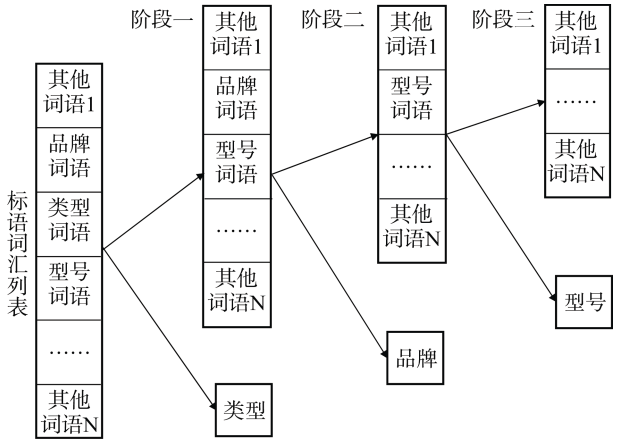

图 12　阶段匹配识别过程

Figure 12　Phase matching

算法 1. 类型匹配算法.

输入: 标语分词后的列表 L_{banner}, 设备类型集合 S_{type}

输出: 过滤设备类型标签 T_{type}, 过滤类型后的标语词汇列表 L_{ftbl}

1. INIT $T_{type} \leftarrow \phi$
2. INIT $L_{ftbl} \leftarrow L_{banner}$
3. $L_{banner} \leftarrow L_{banner}.lower()$
4. $S_{type} \leftarrow S_{type}.lower()$

过程 1.判断 L_{banner} 是否有元素在 S_{type}

5. Get each *element* from L_{banner}
6. IF *element* in S_{type}
7. THEN
8. $T_{type} \leftarrow$ *element*
9. Delete *element* from L_{ftbl}
10. Break
11. ELSE
12. Continue

3.4.2　设备品牌匹配阶段

物联网设备品牌匹配阶段, 是使用品牌库中的品牌与过滤类型后的标语词汇列表中的词进行匹配, 得到设备的品牌, 如算法 2 所示。因为品牌集合中存在大小写, 而匹配算法对大小写敏感, 因而需要将品牌集合全转为小写, 接着把过滤类型后的标语词汇列表 L_{ftbl} 中的每个元素与已有设备品牌集合 S_{brand} 进行比较, 最终得到过滤品牌后的标语词汇列表(L_{ftbbl})以及过滤得到品牌(T_{brand})。

算法 2. 品牌匹配算法.

输入: 过滤类型后的标语词汇列表 L_{ftbl}, 设备品牌集合 S_{brand}

输出: 过滤设备品牌标签 T_{brand}, 过滤类型、品牌后的标语词汇列表 L_{ftbbl}

1. INIT $T_{brand} \leftarrow \phi$
2. INIT $L_{ftbbl} \leftarrow L_{ftbl}$
3. $S_{brand} \leftarrow S_{brand}.lower()$

过程 1.判断 L_{ftbl} 是否有元素在 S_{brand}

4. Get each *element* from L_{ftbbl}
5. IF *element* in S_{brand}
6. THEN

7. $T_{brand} \leftarrow element$
8. Delete *element* from L_{ftbbl}
9. Break
10. ELSE
11. Continue

3.4.3 设备型号匹配阶段

通过 3.2 节自动构建的自动爬取框架, 当前已收集的物联网设备类型有 53 种, 已收集品牌有 2082 个, 所以对于类型和品牌直接使用库匹配过滤, 但是对于型号匹配识别存在如下两个问题:

(1) 由于型号库数量较大, 有 52617 个不同型号, 直接遍历型号库中所有型号与 L_{ftbbl} 中的词汇进行比较, 时间复杂度高;

(2) 即使当前收集了万数级的型号, 但仍存在部分未收集完全的型号。

因此, 本文根据型号库中收集到的型号信息规律, 如表 4 型号规则分析表所示, 构造正则表达式, 通过这部分正则, 可筛选出潜在的设备型号。通过实验分析, 经过正则表达式抽取, 型号匹配时间降低了一倍。同时为了验证正则表达式的科学性, 通过使用正则表达式对所有收集的型号进行过滤验证, 发现已收集的所有型号都满足表 4 对应的规则。

表 4 型号规则分析表

Table 4 Model rule analysis table

型号	对应的设备描述	满足的规则
Micro800	罗克韦尔 PLC 型号	字母数字组合
5070	富士打印机型号	纯数字
DC-0472V-960H	海康威视 DVR 型号	字母数字连线组合

另外, 经过调研后发现, 有的品牌的部分型号名称相似, 直接字符串比较可能会导致部分未收集的型号丢失, 如表 5 所示, 海康威视 DVR 存在极其相似的型号。因此, 本文采用字符串关联算法来匹配识别, 找出更多设备的型号, 并且根据同一系列的设备型号推算出设备的品牌、类型以及类别信息。将这部分潜在的设备型号与型号库中的型号进行相似度匹配识别, 将满足阈值的标语型号输出, 得到设备型号。这样做不仅降低了与型号库比较次数, 提高了整体效率, 更是克服了因为型号库可能收集不全而导致的型号未能正确识别的其他相似型号设备的问题。

为了描述方便, 定义设备型号全集为 S_{model}, 特定类型下的设备型号集合为 S_{tmodel}, 特定品牌下的设备型号集合为 S_{bmodel}, 特定类型和特定品牌下的设备型号集合为 $S_{tbmodel}$ 其中 $S_{tmodel} \subseteq S_{model}$, $S_{bmodel} \subseteq S_{model}$, $S_{tbmodel} \subseteq S_{tmodel}$, $S_{tbmodel} \subseteq S_{bmodel}$。算法具体描述如算法 3 所示, 通过将算法 1 和 2 中得到的设备类别、品牌信息作为输入, 缩小型号库范围, 接着利用正则表达式, 提取 L_{ftbbl} 中潜在的型号, 对这部分潜在型号与型号库中的型号, 调用算法 4 中的型号相似度比较算法, 得到满足阈值的型号信息, 算法 4 中 *LevenshteinRatio* 被用来计算两个字符之间相似度。

表 5 相似型号分析表

Table 5 Similarity model analysis table

型号	设备其他信息描述	备注
DC-0472V-960H	海康威视 DVR 型号	相同品牌不同型号
DC-0472W-960H	海康威视 DVR 型号	相同品牌不同型号

算法 3. 型号匹配算法

输入: 过滤类型、品牌后的标语词汇列表 L_{ftbbl}, 设备型号集合 S_{model}, 设备类型标签 T_{type}, 设备品牌标签 T_{brand}, 相似度匹配阈值 T_{MS}

输出: 过滤设备型号标签 T_{model}

1. INIT $T_{model} \leftarrow \phi$
2. INIT $M \leftarrow S_{model}$
3. INIT $S_{tmodel} \leftarrow \phi$
4. INIT $S_{bmodel} \leftarrow \phi$
5. INIT $S_{tbmodel} \leftarrow \phi$
6. INIT $S_{canmodel} \leftarrow \phi$
7. INIT $S_{RM} \leftarrow M$

过程 1. 根据 T_{type}, T_{brand} 筛选出型号 S_{RM}

8. IF T_{type} and T_{brand}
9. THEN
10. $S_{RM} \leftarrow S_{tbmodel} \leftarrow \mathrm{M}(T_{type}, T_{brand})$
11. ELIF T_{type}
12. THEN
13. $S_{RM} \leftarrow S_{tmodel} \leftarrow \mathrm{M}(T_{type})$
14. ELIF T_{brand}
15. THEN
16. $S_{RM} \leftarrow S_{bmodel} \leftarrow \mathrm{M}(T_{brand})$

过程 2. 正则表达式提取潜在型号

17. Get $S_{canmodel}$ from L_{ftbbl} by the regular expression filter
18. Sorted S_{RM} by desc
19. $L_{sm} \leftarrow \phi$

过程 3. 潜在型号与型号集合元素进行相似度比对

20. Get each *element* from $S_{canmodel}$
21. Get the L_{sm} (<*similarity_1, model_1*>,..., <*similarity_n,model_n*>) tuple from *ModelSimilarity*(*element*, S_{RM}, T_{MS})//过滤掉相似度<T_{MS}的型号
22. IF L_{sm}
23. THEN
24. $T_{model} \leftarrow max(L_{sm}$ (*similarity_1, ..., similarity_n*))

25. ELSE Continue

算法 4. 型号相似度比较算法 ModelSimilarity.

输入: 潜在型号 *candmodel*, 类型、品牌过滤后的型号集合 S_{RM}, 型号相似度门限 T_{MS}

输出: 相似型号列表 L_{sm}

1. INIT $L_{sm} \leftarrow \phi$
2. INIT $low \leftarrow 0$
3. INIT $high \leftarrow Len(S_{RM})$-1

过程 1. 分治法求潜在型号与型号集合元素间的 *LevenshteinRatio* 距离

4. WHILE *low* <= *high*
5. *mid* ← (*low*+*high*)/2
6. *model* ← S_{RM}(*mid*)
7. *similarity* ← *LevenshteinRatio*(*candmodel*, *model*)
8. IF *similarity* > T_{MS}
9. THEN
10. L_{sm} add <*similarity*, *candmodel*>
11. IF *candmodel* == *model*
12. THEN
13. L_{sm} add <*100*, *candmodel*>
14. ELIF *model* > *candmodel*
15. THEN *high* ← *mid* -1
16. ELSE THEN *low* ← *mid* +1

为了选取 T_{MS} 的参数具体的值, 设计 T_{MS} 参数选择实验, 具体的实验步骤如下:

(1) 从型号库中随机选择 20 个不同的型号, 每个型号数量为 10, 总共构成 200 个型号数据集。型号包括的设备类型有 IP Camera、PLC、路由器和打印机, 覆盖的设备品牌有 18 种。此处设每一种系列的型号数据属于一类, 构成的类别向量用 Y 表示, 则 $Y=(y_1,y_2,\cdots,y_{20})$, 型号集合构成 X, $X=(x_1,x_2,\cdots,x_{200})$。设 x_{mn} 为 X 集合中属于 Y 中某一类的型号, n 的取值为 1,2,3,⋯,20, m 的取值为 1,2,3,⋯,200;

(2) 将每一系列的型号字符串进行排序, 即将属于 Y 每一类的 X 进行排序;

(3) 从每一类排好序的型号中随机选择一个型号作为此类的基准型号, 记为 $Q_s=(x_{y1},x_{y2},\cdots,x_{y20})$, 基准型号集中有 20 个型号, 每一个型号都属于 Y 中的不同类;

(4) 将 Q_s 中的每一个型号与 X 中的每一个型号分别进行 *LevenshteinRatio* 距离相似度计算, 得到 Q_s 中的每一类型号与 X 中的每个型号的相似度, 以及每一个相似度下的型号对, 即(x_{yn},x_m),其中 n 的取值为 1,2,3,⋯,20, m 的取值为 1,2,3,⋯,200;

(5) 通过将(4)得到的相似度与 T_{MS} 比较, 若相似度大于 T_{MS}, 则将相似度对应的型号对中的 x_m 归类为 x_{yn}, 记为 x^*_{mn} 从而得到每个 x_{yn} 下的型号分类结果;

(6) 最后根据每个 x_{yn} 下的型号分类结果, 求每个 x_{yn} 类别下的型号分类的准确率和召回率; 其中型号分类准确率用 P_m 表示, 型号分类召回率用 R_m 表示。计算公式如下:

$$P_m = \frac{\sum_m x^*_{mn} \cap \sum_m x_{mn}}{\sum_m x^*_{mn}} \quad (1)$$

$$R_m = \frac{\sum_m x^*_{mn} \cap \sum_m x_{mn}}{10} \quad (2)$$

(7) 计算型号分类准确率的均值和方差, 型号分类召回率的均值和方差;

(8) 设置 T_{MS} 的阈值范围为 50%~100%, 每隔 10%设置一次 T_{MS} 阈值取值, 即 T_{MS} 阈值为[50%, 60%, 70%, 80%, 90%, 100%], 将步骤(5)到步骤(7)重复操作。

得到的实验结果如表 6 所示。为了保证在识别准确的情况下召回更多种类的型号, 因此 T_{MS} 的取值为 90%最合适。

由于 T_{MS} 是在小范围型号数据集上进行的实验, 在整个型号数据集上并不具有说服依据, 因此, 本文选择在整个型号库中进行实验, 验证 T_{MS} 取值的科学性。实验步骤同小范围型号数据集, 最终得到实验结果如表 7 所示。

表 6 相似度阈值小范围取值分析

Table 6 A small range value analysis of similarity degree threshold

T_{MS}	准确率均值	召回率均值	准确率方差	召回率方差
50%	57.55%	76%	0.092	0.085
60%	73.20%	63%	0.085	0.121
70%	83.15%	49%	0.090	0.131
80%	90.50%	32%	0.056	0.063
90%	100%	13.5%	0	0.003
100%	100%	10%	0	0

表 7 相似度阈值大范围取值分析

Table 7 A large range value analysis of similarity degree threshold

T_{MS}	准确率均值	召回率均值	准确率方差	召回率方差
50%	8.20%	94.54%	0.023	0.032
60%	27.23%	92.36%	0.090	0.041
70%	53.04%	88.46%	0.129	0.060
80%	73.70%	82.37%	0.100	0.083
90%	96.90%	70.77%	0.035	0.120
100%	98.68%	66.50%	0.013	0.132

观察发现当 T_{MS} 选择 90%时, 准确率在大范围型号数据集上的准确率只有 96.90%, 而不是小范围得到的 100%, 经过对数据集的分析发现, 存在部分不

同类型或者品牌的型号极其相似, 但由于型号相似比对算法基于 Levenshtein 距离, 因而不同类型或品牌且相似型号计算的距离会大于设定的 T_{MS}。比如 HP 有一款型号为 6000r 的服务器, 同时 IBM 还有一款 6000R 的服务器, 根据 *LevenshteinRatio* 计算导致这 2 款不同品牌的设备有可能被识别为一个产品, 因此导致了在准确率方面略有下降, 但该准确率在可接受范围内。与 T_{MS} 取 100%相比, 发现了虽然准确率略有降低, 但召回率得到了提升, 证明在大范围数据集上, 本方法可以有效找到更多相似的型号。综合以上原因, 继续选择 T_{MS} 为 90%。

3.4.4 验证反馈

数据预处理过程中的关键字提取部分的过滤规则库是根据初始的设备协议标语信息样本集合提取出来的。但通常情况下能够接触的样本集合也比较有限, 根据少量的样本集合获取得到的过滤规则库通常是有偏的, 不全面的, 这样最终也会影响设备匹配识别的准确率。例如, 表 8 给出的是 FTP 协议原始标语, 通过最初始的设备过滤规则库, 在经过数据预处理之后得到的标语词汇列表如表 9 所示。然后通过匹配识别之后, 会将此标语识别成型号为 5070 的富士打印机。但实际上根据此 FTP 协议标语是无法获取设备的品牌和型号。通过观察发现, “You are user number N of M”是 FTP 协议标语返回格式的一种, 其目的是告诉访问者的顺序, 因而此语句中的字符不能用来区分设备的品牌或型号, 反而会影响到最终的设备型号匹配识别的准确率。

表 8 FTP 协议标语原始信息样例

Table 8 Raw FTP protocol banner

FTP 协议标语原始信息样例
\r\nYou are user number 5070 of 10000 allowed. 220 ucftpd(Sep 10 2010-17:23:34) FTP server ready.\r\n.

表 9 预处理后的 FTP 协议标语信息样例

Table 9 Preprocessed FTP protocol banner

预处理后 FTP 协议标语词汇列表样例
You are user number 5070 10000 allowed ready

经过上述验证发现, 需要将 FTP 协议标语中的“You are user number N of M”这句话过滤, FTP 协议标语的过滤规则库需要添加一条新的规则。

为了验证反馈阶段对识别方法的影响, 本次实验利用了通用协议标语——FTP 协议标语数据集通过如下步骤进行验证:

(1) 数据获取, 通过 Zgrab[20]探测工具对全网开放 FTP 协议的设备进行标语抓取, 得到超过 2000 万的设备 FTP 协议标语信息;

(2) 识别匹配, 使用品牌型号库设备识别模块对 FTP 协议标语数据进行处理和识别;

(3) 结果统计, 对识别的结果进行统计;

(4) 抽样验证, 抽样验证识别结果的准确性(此处主要是对型号的识别准确率进行抽样验证);

(5) 根据验证结果, 判断识别型号错误的原因;

(6) 更新过滤规则库的规则;

(7) 再次识别, 重复步骤(2)—(6), 直至规则无法进一步添加。

在使用品牌型号库识别方法对 FTP 协议标语识别过程中, 新增了两条规则, 并且将此方法在FTP协议标语的设备型号识别准确率从 64.8%提高到了 94%。如上文所述, 规则和协议之间存在映射关系。其中新增的两条规则如下:

规则 1: 若设备 FTP 协议标语信息中存在“You are user number N of M”,在数据预处理阶段将此句话删除, 如表 10 所示。在全网中, 设备 FTP 协议标语信息中存在此语句的标语有 3165393 条, 而错误的将此语句的数字识别成型号的有 659945 条, 占比高达 21%; 增加此规则之后, 可以将型号准确率提高到 84%。

规则 2: 根据品牌型号库算法模型的识别方法, 若在前两阶段都未识别出设备的产品属性, 则在第三阶段即型号匹配识别阶段, 增加一个过滤数字字符串过程。其意义是根据设备 FTP 协议标语中进行了规则 1 过滤之后, 仍然存在 26 万左右的数据其标语中含有数字字符串, 但是不能代表设备的型号, 因此需要将其过滤。增加规则 2 之后, 可以将型号准确率从 84%提高到 94%。表 11 展示了人工验证反馈对型号准确率的影响。

表 10 规则处理后的 FTP 协议标语信息样例

Table 10 FTP protocol banner after regularization processing

FTP 协议标语原始信息样例
\r\n220 ucftpd(Sep 10 2010-17:23:34) FTP server ready.\r\n.

表 11 人工验证反馈影响

Table 11 Feedback effect of manual verification

规则条件	初始规则	增加规则 1	增加规则 2
型号准确率	64.8%	84%	94%

4 框架性能评估

由于物联网设备品牌型号总数较多, 为了验证本文提出的框架对品牌型号识别覆盖度情况, 分别

利用框架与人工指纹在 4.1 节进行品牌型号识别覆盖度验证实验。算法 4 提出了正则过滤型号的方法, 为了验证该方法有效性, 在 4.2 节设计了正则抽取对算法效率的影响实验。最后, 为了验证整体框架识别的准确性与细粒度的情况, 在 4.3 节设计了 2 种视频监控系统协议 Onvif、FTP 的型号准确率验证实验, 在 4.4 节设计了 2 种工控系统协议 Ethernet/ip、Bacnet 的型号准确率验证实验。

4.1 品牌型号识别覆盖度验证

为了验证基于搜索的设备产品属性识别框架是否能够召回更多设备品牌和型号, 本次实验利用了全网扫描探测的部分协议标语数据集对品牌和型号种类覆盖度进行了验证, 具体的验证步骤如下:

(1) 数据获取: 对 20 个工控专有协议、3 个视频监控专有协议、3 个打印机专有协议和 6 个通用协议在全网空间进行标语信息抓取;

(2) 识别匹配: 使用基于搜索的设备识别框架以及人工收集指纹分别对上述 32 个协议标语信息数据集分别识别;

(3) 结果统计: 统计步骤(2)的识别结果, 将识别结果按照类别、类型、品牌和型号聚类;

(4) 结果分析: 通过对聚类结果分析可知, 经过 4 个月自动化构建的品牌型号库识别出的物联网设备品牌种类超过了 1200 种, 识别的物联网设备型号种类超过了 12000 种, 识别的数量远远大于 2 年收集到的人工指纹库的 170 种设备品牌, 1843 种设备型号。

通过品牌型号库的识别方法, 能够有效的发现网络空间中的设备品牌和型号。与人工指纹收集相比, 本方法极大地提高了设备识别的覆盖能力。

4.2 正则抽取对算法影响

由于型号库中的型号数量庞大, 达到万数量级, 为了能够达到快速匹配识别的效果, 故对已经过滤完品牌和类型的标语数据列表 L_{ftbbl} 进行了正则抽取过程, 进一步过滤掉非型号相关词汇。

为了验证在型号匹配识别之前加入正则抽取过程的确能够节约时间, 故据此设计了一个对照实验, 通过对照实验进行验证。实验步骤如下:

(1) 数据获取: 通过从 Censys 中下载全网的 FTP 协议标语数据、SSH 协议标语数据和 Http 协议标语数据;

(2) 数据过滤: 通过本章提到的非物联网设备过滤方法分别对步骤(1)中的三个协议标语数据过滤, 分别得到三个协议标语数据的物联网设备数据集; 保证得到的数据集在使用库识别方法识别中, 一定会经过类型、品牌和型号匹配识别阶段;

(3) 实验数据抽取: 从步骤(2)中得到的三个协议标语的数据集分别抽取 2 万条协议标语数据作为实验数据, 即 2 万条 Http 协议标语数据集Ω_1、2 万条 FTP 协议标语数据集Ω_2和 2 万条 SSH 协议标语设备数据集Ω_3;

(4) 对照实验: 分别准备库匹配算法的两套程序 A 和 B 进行实验, 其中 A 是在库匹配算法中使用了正则抽取模块、B 是没有使用正则抽取模块的库匹配算法。

使用三组数据集分别对 A 和 B 进行测试, 计算 A 和 B 在完成三组数据集完整的识别过程中各自消耗的时间。

实验结果如表 12 所示。从实验结果中可以看出, 利用正则抽取方法对 L_{ftbbl} 进行过滤可以节省更多的匹配时间, 即在库匹配算法中使用正则抽取, 可以让整个品牌型号库识别方法达到更快的识别效果。通过实验结果数据可知, 不使用正则抽取整个识别方法的时间是使用正则抽取所花的时间 2 倍到 9 倍左右, 因此正则抽取可以极大的提高算法的性能。

表 12 正则抽取对算法性能影响

Table 12 Regular extraction effects on algorithm performance

数据集名称	数据集大小	A 耗费的时间(s)	B 耗费的时间(s)	B/A
Ω_1	20000	218	500	2.29
Ω_2	20000	52	518	9.96
Ω_3	20000	23	136	5.91
均值		97.67	284.67	6.05

4.3 视频监控设备型号识别

为了验证物联网设备识别框架对互联网视频监控设备识别的有效性, 通过如下步骤进行验证:

(1) 数据获取: 选取视频监控设备 2 个常用协议 Onvif 与 FTP, 进行全网标语抓取, 得到 Onvif 协议标语 813404 条, FTP 协议标语 13110691 条;

(2) 识别匹配: 使用本文提出的框架, 对得到的 Onvif 与 FTP 协议标语信息进行识别, 得到识别结果, 其中被识别为视频监控设备的数据分别为 305622 与 12302 条;

(3) 数据集压缩: 对识别出是视频监控设备的 2 个协议数据进行压缩, 去除重复标语信息后, 得到 Onvif 协议覆盖型号 3441, FTP 协议覆盖型号 378 种;

(4) 验证结果: 通过对(3)去重之后的数据进行型号识别准确率验证, 得到的设备型号识别准确率

分别为 97%与 90.87%。

通过对以上 2 种不同视频监控设备常用协议进行识别, 如表 13 所示, 可以发现本文提出的物联网设备识别框架, 对不同协议可以达到不同型号识别准确率, 最高可以达到 97.00%, 最低可以达到 90.87%。一定程度上可以说明本文提出的框架的确可以达到细粒度对视频监控设备的识别。

表 13 Onvif 与 FTP 标语 VSS 识别结果分析
Table 13 VSS recognition results analysis of Onvif and FTP

协议类别	识别出视频监控设备的数据	涵盖型号的种类	设备型号识别准确率
Onvif	305622	3441	97.00%
FTP	12302	378	90.87%

4.4 工控设备型号识别

为了能够验证本识别框架对工业物联网环境下的工控设备识别的有效性, 本次实验利用工控设备专有协议标语——Ethernet/ip 与 Bacnet 通过如下步骤进行验证:

(1) 数据获取: 通过对全网开放 Ethernet/ip 与 Bacnet 协议的设备进行标语抓取, 得到设备 Ethernet/ip 协议标语数据 8759 条, Bacnet 协议标语数据 12203 条;

(2) 识别匹配: 使用本文提出的方法分别对得到的协议标语数据进行识别, 得到识别结果, 其中 Ethernet/ip 识别出工控设备的数据为 5811 条, Bacnet 识别出工控设备的数据为 7361 条;

(3) 数据集压缩: 对识别出是工控设备的数据进行压缩, 去除重复的标语信息后, 统计发现 Ethernet/ip 协议覆盖型号种类 290 种, Bacnet 协议覆盖型号种类 220 种;

(4) 验证结果: 通过对(3)去重之后的数据进行型号识别准确率验证, 得到的设备型号识别准确率分别为 97.00%与 97.77%。具体内容如表 14 所示。

通过对以上 2 种不同工控系统设备专有协议进行识别分析, 本文提出的物联网设备识别框架对工业物联网环境下的工控设备识别仍然适用, 同时还能达到较高的型号识别准确率。

本方法不仅仅支持 Ethernet/ip 与 Bacnet 协议, 同时还支持其他 18 种的工控设备专有协议标语, 但 Shodan 目前仅支持 15 种。本方法与物联网设备搜索引擎 Shodan 在全网工控设备识别数量上比较如图 13 所示。图 13 显示了 11 种不同工控设备数量对比, 除了 fox 协议识别数量比 Shodan 少之外, 剩余 10 种都是领先或者与 Shodan 持平。出现 fox 协议识别数量少于 Shodan 的现象, 有很大可能性是因为探测系统对 fox 协议探测数据包收集还不够全, 导致识别出的 fox 设备数量稍微低于 Shodan。

表 14 Ethernet/ip 与 Bacnet 标语 ICS 识别结果分析
Table 14 Analysis of ICS recognition results of Ethernet/ip and Bacnet banners

协议类别	识别出工控设备的数据	涵盖型号的种类	设备型号识别准确率
Ethernet/ip	5811	290	97.00%
Bacnet	7361	220	97.77%

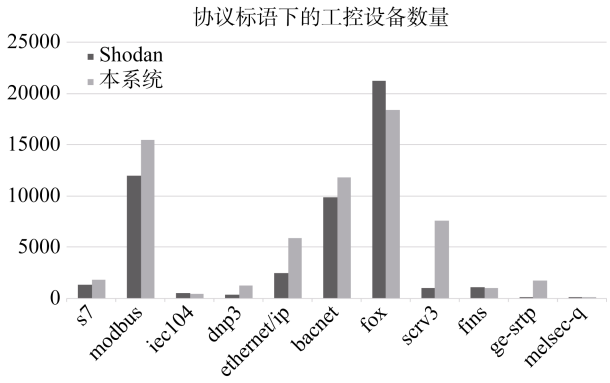

图 13 工控协议下的设备数量分析
Figure 13 Number of recognizable ICS device statistics

5 未来工作

本文提出了一种基于搜索的物联网设备产品属性识别框架。收集了 2397 种的物联网设备品牌种类, 56282 种物联网设备型号种类, 确定了 10 种设备类别, 目前可识别的物联网品牌种类达到 1200 种以上, 可识别物联网型号种类达到 12000 种以上。通过设计算法成功实现对物联网设备协议标语信息内容的提取工作。本文还对公网上的视频监控设备系统以及工业控制系统进行了实验。实验表明, 本框架在对通用、专用协议, 视频监控和工控设备都有较好表现, 产品属性识别准确率均超过 90%。

实验表明, 尽管论文提出方法能够实现对设备产品属性可以进行准确识别和标定, 但在设备识别率方面并不高。实验表明, 互联网可探测存活设备有 4~5 亿, 物联网设备根据推断在 1 亿左右(包含家用路由器), 而当前能识别的设备不超过 5000 万。主要问题包括:

(1) 大量设备在进行协议标语抓取过程中受到防火墙等防护设备的拦截, 导致很大一部分开放端口不能抓到正常返回的标语信息;

(2) 随着安全意识的提升, 大量物联网设备都在提供服务信息前增加统一认证过程, 仅通过标语信息难以区分不同设备;

(3) 部分探测数据包收集不够完善, 导致探测获取原始标语信息的数量不全。

未来将在探测的友好度、探测数据包的完整性以及探索除上层协议标语外的其他设备特征提取技术和利用方法方面进行改进, 以提升对物联网设备识别比例。

参考文献

[1] "Gartner. (2017) Trend prediction of iot devices" Gartner, https://www.gartner.com/newsroom/id/3598917, Feb. 2017.

[2] "Common Vulnerabilities and Exposures" CVE, http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2015-7254, Sept. 2015.

[3] "Shodan. The search engine for Internet-connected devices." https://www.shodan.io/

[4] Z. Durumeric, A. David, M. Ariana, B. Michael, and J.A. Halderman, "A search engine backed by Internet-wide scanning," In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*(CCS'15), pp. 542-553, Oct. 2015.

[5] T. Kohno, A. Broido, and K.C. Claffy, "Remote physical device fingerprinting," *IEEE Transactions on Dependable & Secure Computing*(TDSC'05), pp. 93-108, Apri. 2005.

[6] A. Cui, and S.J. Stolfo, "A quantitative analysis of the insecurity of embedded network devices: results of a wide-area scan," In *Proceedings* of the *26th Annual Computer Security Applications Conference*, pp. 97-106, Dec. 2010.

[7] S.V. Radhakrishnan, A.S. Uluagac, and R. Beyah, "GTID: A technique for physical device and device type fingerprinting," *IEEE Transactions on Dependable and Secure Computing*(TDSC'15), pp. 519-532, Sep. 2015.

[8] L.C. Cao, J.J. Zhao, X. Cui, and K. Li, "Cyberspace device identification based on K-means with cosine distance measure," *Journal of University of Chinese Academy of Sciences*, vol. 33, no. 4, pp. 562-569(in Chinese), 2016.
(曹来成, 赵建军, 崔翔, 李可, "基于余弦测度下 K-means 的网络空间终端设备识别[J]", *中国科学院大学学报*, 2016, 33(4): 562-569。)

[9] L.C. Cao, J.J. Zhao, X. Cui, and K. Li, "Cyberspace Terminal Device Identification Framework," *Computer systems and Applications*, vol. 25, no. 9, pp. 60-66(in Chinese), 2016.
(曹来成, 赵建军, 崔翔, 李可, "网络空间终端设备识别框架[J]", *计算机系统应用*, 2016, 25(9): 60-66。)

[10] R.L. Ren, Y. Gu, J. Cui, S. Liu, H.S. Zhu, and L.M. Sun, "Web Features-based Recognition Specific-Type IoT Device in Cyberspace," *Communications Technology*, vol. 50, no. 5, pp. 1003-1009(in Chinese), 2017.
(任春林, 谷雨, 崔杰, 刘松, 朱红松, 孙利民, "基于 web 信息的特定类型物联网终端识别方法", *通信技术*, 2017,50(5): 1003-1009。)

[11] M. Miettinen, S. Marchal, I. Hafeez, N. Asokan, A.R. Sadeghi, and S. Tarkoma, "Iot SENTINEL: automated device-type identification for security enforcement in iot," In *Distributed Computing Systems* (ICDCS'17), pp. 2177-2184, June. 2017.

[12] Q. Li, X. Feng, H.N. Wang, and L.M. Sun, "Automatically Discovering Surveillance Devices in the Cyberspace." In *Proceedings of the 8th ACM on Multimedia Systems Conference*(MMSys'17), pp. 331-342, Jun. 2017.

[13] Q. Li, X. Feng, Z. Li, H.N. Wang, and L.M. Sun, "GUIDE: Graphical user interface fingerprints physical devices," in *IEEE International Conference on Network Protocols*(ICNP'16), pp. 1-2, Nov. 2016.

[14] Y. Meidan, B. Michael, S. Asaf, J.D. Guarnizo, O. Martín, N.O. Tippenhauer, and E. Yuval, "ProfilIoT: a machine learning approach for IoT device identification based on network traffic analysis," In *Proceedings of the Symposium on Applied Computing*, pp. 506-509. April, 2017.

[15] B. Bezawada, M. Bachani, J. Peterson, H. Shirazi, I. Ray, and I. Ray, "IoTSense: Behavioral Fingerprinting of IoT Devices," *arXiv preprint arXiv*:1804.03852, Apr. 2018.

[16] F. Shaikh, E. Bou-Harb, J. Crichigno, and N. Ghani. "A Machine Learning Model for Classifying Unsolicited IoT Devices by Observing Network Telescopes," In *IEEE International Wireless Communications and Mobile Computing Conference*. IEEE, 2018.

[17] M. Bristow, "ModScan: a SCADA MODBUS Network Scanner," in *DefCon-16 Conf*, 2008.

[18] "Tool for scan PLC devices over s7comm or modbus protocols," https://code.google.com/archive/p/plcscan/, Sept, 2012.

[19] X. Feng, Q. Li, Q. Han, H.S. Zhu, Y. Liu, and L.M. Sun, "Identification of visible industrial control devices at internet scale," in *IEEE International Conference on Communications* (ICC'16), pp. 1-6, May, 2016.

[20] "Zgrab. A stateful application-layer scanner that works with ZMa-p." https://github.com/zmap/zgrab, Oct, 2014.

邹宇驰 于2016年在中国矿业大学网络工程专业获得学士学位。现于中国科学院信息工程研究所信息安全专业攻读硕士学位。研究领域为设备识别技术、网络探测。研究兴趣包括: 物联网安全、网络空间测量。Email: zouyuchi@iie.ac.cn

刘松 于 2015 年在太原理工大学物联网工程专业获得学士学位, 现于中国科学院大学信息工程研究所计算机技术专业攻读硕士学位, 研究领域为设备识别技术, 研究兴趣包括:物联网安全, 网络空间测量。Email: zoe_liu0520@126.com

于楠 于 2011 年在北京航空航天大学获得硕士学位。现于中国科学院信息工程研究所任助理研究员, 研究领域为物联网安全和计算机网络, 研究兴趣包括大数据存储、传输与分析。Email: yunan@iie.ac.cn

朱红松 于 2009 年在中国科学院大学计算所获得博士学位。现任中国科学院信息工程研究所研究员。主要研究方向包括物联网安全、网络攻防、安全大数据分析与测评。Email: zhuhongsong@iie.ac.cn

孙利民 于 1998 年在国防科学技术大学计算机学院获得博士学位。现任中国科学院信息工程研究所研究员。主要研究方向为物联网及其安全、工业控制系统安全、区块链安全。Email: sunlimin@iie.ac.cn

李红 于 2017 年在中国科学院大学信息安全专业获得博士学位。现任中国科学院信息工程研究所 助理研究员。主要研究方向为物联网安全、区块链安全。Email: lihong@iie.ac.cn

王旭 于 2017 年在西安电子科技大学电子信息工程专业获得学士学位。现于中国科学院信息工程研究所网络空间安全专业攻读硕士学位。研究领域为联网设备探测与识别, 研究兴趣包括深度学习, 计算机网络。Email: xuwang.me@gmail.com