网络空间物联网信息搜索

李 强[1], 贾煜璇[1], 宋金珂[1], 李 红[2,3], 朱红松[2,3], 孙利民[2,3]

[1] 北京交通大学 计算机与信息技术学院 北京 中国 100044
[2] 中国科学院大学 网络空间安全学院 北京 中国 100049
[3] 中国科学院信息工程研究所 北京 中国 100093

摘要 本篇论文总结和分析了网络空间物联网信息搜索相关研究工作, 作为物联网信息搜索的综述性工作。物联网信息是网络空间中最重要的资产, 在各个领域行业发挥着越来越重要的作用。探测、发现和识别网络空间中的物联网信息, 已经成为了保障网络空间关键基础设施安全的前提和有效手段。本文, 首先提出了网络空间物联网信息搜索的基本架构。其次, 论文讨论了四类典型物联网信息的相关研究工作, 包括操作系统信息、应用服务、设备种类和标识信息。网络空间存在着海量、动态和异构的物联网信息, 本文总结和分析了物联网信息搜索关键技术的研究, 包括探测技术和识别技术。最后, 论文探讨了两类基于物联网信息搜索的应用, 包括互联网空间测量和大规模安全事件分析。

关键词 物联网; 网络空间; 物联网信息
中图法分类号 TP309.1 **DOI 号** 10.19363/J.cnki.cn10-1380/tn.2018.09.04

Search of Internet of Thing Information in the Cyberspace

LI Qiang[1], JIA Yuxuan[1], SONG Jinke[1], LI Hong[2, 3], ZHU Hongsong[2, 3],SUN Limin[2, 3]

[1] School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China
[2] School of Cyber Security, University of Chinese Academy of Science, Beijing 100049, China
[3] Institute of Information Engineering, Chinese Academy of Science, Beijing 100093, China

Abstract In this paper, we have provided the overview of Internet-of-things (IoT) information search in the cyberspace. Nowadays, IoT devices have become the most critical asset in cyberspace and played an increasingly important role in various industries, including smart grid and industrial control systems. In both offensive and defensive perspectives, discovering IoT information would stays the core position in the cyberspace security. In this paper, we first propose the overall architecture for searching IoT information. Secondly, we have detailed the typical IoT information, including operate systems, application services, device types and identifier information. Third, we have demonstrated key technologies in the research of IoT search, including detection and identification technology. Finally, we present the cyberspace security application based on IoT search.

Key words Internet of Things Information ; Cyberspace; IoT Devices

1 引言

网络空间是相互连接的信息系统基础设施所形成的人造空间, 人们在网络空间开展各类相关活动并传递各类信息。它是互联网(Internet)、电信网络、广域网和局域网等基础信息设施构建的相互依存的网络。随着物联网在社会各个领域的广泛应用, 越来越多的物联网信息分布在网络空间, 包括操作系统、应用服务和物理设备等等。据 Gartner 报道[1], 每天都有上百万的物联网设备接入网络空间, 预计在 2020 年接入网络空间的物联网设备将接近 250 亿。物联网信息已经成为了网络空间的重要资产。

物联网融入网络空间的同时, 也暴露了许多安全问题。2016 年 10 月 21 日, 黑客利用大量被 Mirai 病毒感染的物联网设备, 发动了针对 Dyn 管理 DNS 服务器的 DDoS 攻击, 影响范围涵盖了美国东海岸、西海岸和欧洲部分地区, 导致 Twitter、GitHub、亚马逊、PayPal、BBC、华尔街日报等很多知名网站无法

通讯作者: 李红, 职称: 助理研究员。
本课题得到国家重点研发计划(No. 2018YFB0803402), 国家自然科学基金重点(No.U1766215), 国家自然科学基金(No.61602029)资助。
收稿日期: 2018-02-26; 修改日期: 2018-06-04; 定稿日期: 2018-08-20

访问。研究人员 Antonakakis[2]分析了感染 Mirai 病毒的物联网设备, 提取了相关的设备种类信息, 包括设备类型、厂商和相关品牌型号。因此, 一旦出现物联网设备被恶意滥用, 受到破坏性攻击, 物联网信息泄露等安全问题, 都会为整个网络空间的安全带来了巨大风险和挑战。2016 年 4 月 19 日, 习总书记在网络安全和信息化工作座谈会上强调了金融、能源、电力、通信、交通等领域的关键信息基础设施在经济社会运行中的重要性, 保障关键信息基础设施的安全是网络安全的重中之重, 研究网络空间物联网的搜索则是保障关键信息基础设施安全的前提。

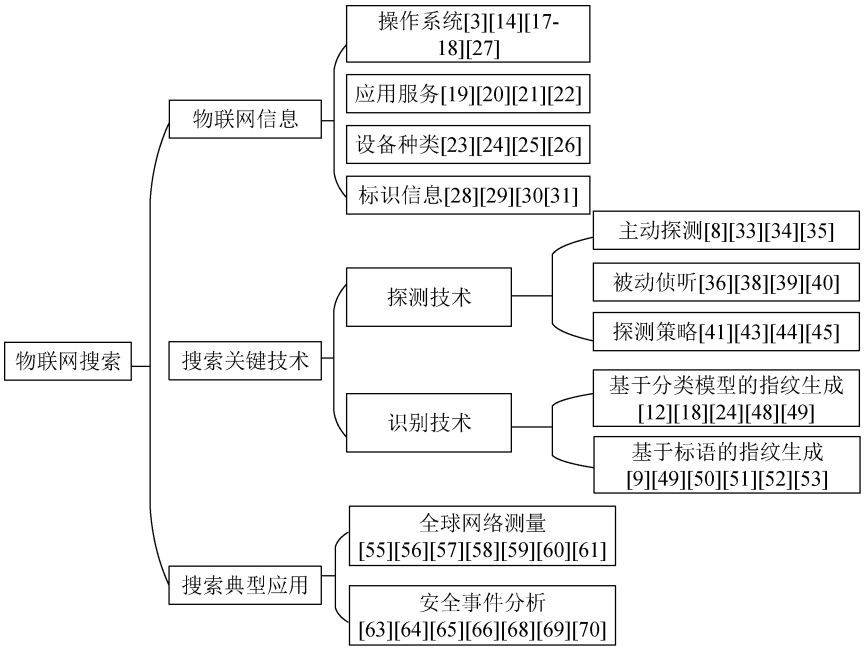

图 1　物联网搜索框架

Figure 1　Framework of Internet of Things Search

近几年, 网络空间物联网信息搜索成为了工业界和学术界的研究热点。工业界, J. Matherly 首次发布 Shodan[4]网络设备搜索引擎。Shodan 是物联网搜索的重要里程碑, 发现了互联网上大量的工控设备和监控设备。学术界, 密西根大学的安全团队推出了 Censys[5]搜索引擎。Censys 利用网络空间快速探测工具 Zmap, 每个星期更新一次 IPv4 空间主流的网络协议, 包括 HTTP, FTP, Telnet 等应用层协议。Censys 结合设备识别工具 Ztag 来识别网络空间中的工业控制系统以及运行 Modbus 工业控制协议的设备[6]。

网络空间物联网信息搜索, 即探测和识别物联网相关信息。传统搜索(谷歌、百度)是基于网络爬虫, 抓取网页内容, 利用文本分析技术结合排序算法建立高效索引, 为用户提供查询服务。物联网信息搜索不同于传统的信息搜索技术, 采用网络探测技术, 结合物联网指纹, 发现和识别网络空间中的物联网信息。

网络空间物联网信息搜索的相关技术研究已经有十几年的发展历史。网络空间物联网信息搜索主要涉及两个关键技术, 探测技术和识别技术。研究工作[7]总结了近 20 年探测技术的发展历史, 分析了探测技术面临的问题和挑战。Nmap[3]是最早的网络探测工具, 通过发送探测包和收集响应包, 完成网络空间探测任务。Nmap 的探测范围比较全面, 但探测速度较慢。近年来, 密西根大学研究团队提出的 Zmap [8]极大地提高了网络探测速度, 可以在数十分钟内完成 40 亿的 IPv4 地址空间的探测。Zmap 的探测速率是 Nmap 探测速率的数千倍。Xuan[9]提出了 ARE 原型系统, 在 Zmap 搜索的结果上, 自动生成物联网信息。J.Richard[10]的综述工作总结了探测技术在网络安全检测方面的应用和发展。识别技术是基于探测技术收集的数据, 提取物联网相关信息, 例如操作系统、应用服务、物理设备种类和标识信息。研究人员 Zain [11]提出了利用 TCP 数据包重传超时时延来识别网络空间中的操作系统版本信息, T.Kohno [13]提出了利用设备的时钟偏移提取物联网设备的标识信息。目前国内外缺少对物联网信息识别技术的综述性工作。

针对网络空间物联网信息搜索的关键技术, 本篇论文综述性地分析和总结了相关研究工作进展, 并阐

述了它所面临的问题与挑战。图 1 描述了物联网信息搜索研究进展的相关工作总结。论文从三个方面分析物联网信息搜索: 物联网信息种类, 关键技术和基于物联网信息搜索的典型应用。首先, 论文阐述了四类典型的物联网信息: 操作系统信息、应用服务信息、设备种类信息和标识信息。文献[3, 14, 17-18, 27]从探测时间和精度方面, 介绍了操作系统信息搜索的研究工作; 文献[19-22]基于通用应用层协议和专有协议, 分别阐述应用服务搜索的研究工作; 文献[23-26]描述了提取设备种类信息的研究工作, 包括设备类型、厂商和品牌型号信息; 文献[28-31]阐述了基于物理信息标识的研究工作。其次, 物联网信息搜索的关键技术包括探测技术和识别技术。论文从主动探测、被动侦听和探测策略等三个方面, 分析探测技术的优缺点。文献[8, 33-40]讨论了主动探测和被动侦听的优缺点; 基于探测顺序、探测服务器和探测包数目, 文献[41-45]阐述了探测策略的相关研究工作。识别技术包括基于分类模型的指纹生成技术和基于标语的指纹生成技术。基于分类模型的指纹生成技术[12, 18, 24, 48-49]需要采集训练数据提取特征值, 利用学习算法生成识别物联网信息的指纹。标语识别技术[9, 49-53]需要提取物理设备的硬编码信息, 生成识别物联网信息的指纹。最后, 论文介绍了基于物联网信息搜索的典型应用: 互联网空间测量[55-61]和大规模安全事件分析[63-70]。本篇论文是第一个总结网络空间物联网信息搜索研究进展的综述性文章。

论文的组织结构如下: 章节 2 提出了物联网信息搜索的总体架构, 章节 3 阐述了网络空间中典型的物联网信息, 章节 4 总结和分析了物联网信息搜索的关键技术, 章节 5 阐述了基于物联网信息搜索的应用, 第六章是论文的总结和展望。

2 物联网搜索的框架

物联网信息搜索, 通过布置探测器, 采取主动或被动的探测技术, 结合探测策略, 收集网络空间中的相关数据, 结合物联网信息的识别技术, 生成物联网信息的相关指纹, 从而提取网络空间中的物联网信息。网络空间, 论文基于已有的学术定义“在相互连接的信息系统基础设施的人造空间, 人们在网络空间开展各类相关活动并传递各类信息”, 包含了互联网(Internet)、电信网络、广域网和局域网。

本篇论文提出了网络空间物联网信息搜索的架构, 如图 2 所示。物联网信息搜索架构主要包括三个部分: 探测、识别和应用。物联网信息探测主要收集网络空间中的传输层信息、网络层信息和应用层信息。根据数据收集方式的不同, 探测技术可以分为主动探测和被动侦听; 根据探测器、探测报的数目和探测顺序的不同, 探测技术可以采用不同的策略去探测网络空间。物联网信息识别, 基于探测技术收集的数据, 识别模块发现和提取网络空间中的物联网信息。物联网信息的指纹, 基于生成指纹方式的不同, 分为基于分类模型的物联网指纹和基于标语的物联网指纹。物联网信息识别, 通过匹配网络空间中的探测数据和指纹, 从而提取相关的物联网信息, 完成物联网信息搜索。应用模块, 基于海量的物联网信息, 确定数据分析的方法, 采用具体的数据存储方式建立数据索引和构建可视化界面, 提供可视化物联网信息。本文主要介绍两类基于物联网搜索的典型应用: 大规模安全事件分析和互联网空间测量。研究人员可以根据物联网信息搜索架构(图 2)搭建原型系统, 建立基于物联网信息搜索的大规模安全事件分析和量化分析的模型, 发现网络空间中的潜在隐患。

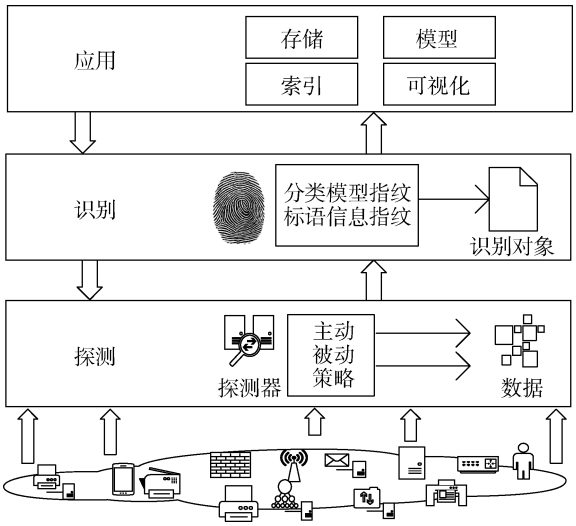

图 2 物联网信息搜索架构

Figure 2 Architecture of Internet of Things Search

3 物联网信息

本章节从操作系统信息、应用服务信息、设备种类和设备标识信息四类典型的物联网信息。

3.1 操作系统信息

操作系统信息是, 作为典型的物联网信息, 是指运行的操作系统版本和型号。识别技术的研究早期主要集中于操作系统识别, 随后逐渐扩展到应用服务、设备类型和标识等方面。操作系统信息不仅包含运行在个人计算机、服务器, 还包含许多小型的嵌入式设备, 例如路由器、打印机、摄像头。Li[27]分析了数千种种嵌入式设备固件, 发现大部分的固

件采用简化版本 Linux 操作系统。这些嵌入式设备的操作系统运行在设备固件上。设备固件是基于 ARM 或者 MIPS 架构, 运行在设备的只读内存区域, 一旦设备运行, 固件会自动加载。

操作系统是基于 TCP/IP 协议栈实现上的差异性来进行识别, 其研究重点集中在降低识别时间和提高识别精度两个方面。Nmap[3]是最早来识别操作系统的研究工具, 它采用 15 个探测包组合(其中包括 12 个 TCP 报文、1 个 UDP 报文和 2 个 ICMP 报文), 基于回应包头字段内容(TCP/IP 协议栈的报文初始化大小、TTL 值、TCP 滑动窗口大小、最大分段长度等特征)的差异性来构建不同操作系统版本的指纹。Nmap 识别操作系统信息有两个限制: (1)随着操作系统版本种类的增多, 识别精度会下降; (2)探测包的数目较多, 仅适合设备数量有限的操作系统识别。Xprobe[14]利用不同探测包获取信息的差异来选取探测包, 并优化重排发送序列, 使用少量探测包来识别操作系统, 但识别精度也有所降低。P0f[15]采用被动侦听的方式, 不发送任何探测包, 仅分析 TCP/IP 报文和数据流量, 其识别精度和速度依赖于探测器的布置情况。SinFP[16]结合了主动探测和被动侦听, 仅采用少量正常的 TCP 报文进行探测, 通过被动侦听到的报文来识别操作系统, 在发送少量报文的情况下实现了操作系统的识别。研究人员 Z.Shamsi[11]提出了 Heshel, 采用发送单个数据包(TCP-SYN)和被探测对象生成的重传(TCP-ACK), 提取数据包间的重传时延, 识别操作系统信息。单数据包的识别技术, 极大地缩短了操作系统信息的搜索时间, 适应于大规模的网络探测, 但也面临着网络抖动、丢包等带来的精确度不高的问题。研究人员[17]提出了一个无参数的 EM 算法来确定重传超时延迟的分布和概率条件, 分析了网络丢包和抖动对超时重传时延的影响, 得到单数据包识别精度的上界。研究人员 David[18]提出了基于机器学习的操作系统识别面临着与 Nmap 同样的问题, 当识别种类增多时, 性能明显下降。

在识别时间上, 大规模操作系统信息搜索只能识别粗粒度版本信息。在识别精度上, 随着操作系统版本种类的增多, 识别准确率会下降。

3.2 应用服务

应用层服务运行在物联网设备上, 提供远程访问和配置管理等基本的功能。设备厂商开发这些应用服务, 并嵌入到物联网设备的只读存储器(ROM)中。应用层的服务信息包含了物联网设备的相关信息。应用层服务信息包括基于通用应用协议的服务和基于专有协议的服务。表格 1 描述了常见的通用协议和专有协议。如果通用协议和专有协议运行在 TCP/IP 协议栈, 那么, 探测技术就能收集相关的应用服务数据, 识别技术提取应用服务信息。本章节分析能够远程获得的应用服务信息的研究工作。

表 1 应用层协议

Table 1 Application Layer Protocol

	协议种类	相关研究工作
通用协议	HTTP, FTP, Telnet, SSH, NTP, RTSP	[3] [14] [19] [20]
专有协议	工业控制协议(Modbus, Siemens S7 等), Onvif	[3] [21] [22]

通用应用层协议, 包括 HTTP, FTP, Telnet 等等。一般来说, 应用层协议和端口进行绑定来实现应用服务端到端之间的通信。端口信息采用 16 比特来描述, 在 0 ~ 65536(2^{16})之间。通用应用层协议运行在固定的端口, 例如 HTTP 默认开放 80 端口, FTP 协议开放 21 端口, Telnet 开放 23 端口。目前常用的工具 Nmap[3]和 Xprobe[14]都支持通用应用层协议相关信息的提取。基于 TCP 的应用服务, 首先需要建立三次握手协议, 然后发送应用层探测包获取服务相关的信息; 基于 UDP 的应用服务, 服务器得到 UDP 的响应包后再发送探测包, 获取服务相关的信息。Web 服务是网络空间中最常见的应用层服务, WhatWeb[19]和 Wapplyzer[20]是提取 web 应用服务信息的典型工具。WhatWeb 拥有 1000 多个插件, 每一个插件能识别一项 web 服务的信息。Wapplyzer 提供了开源代码, 采用正则表达式去提取 Web 服务的相关信息。Nmap 和 Xprobe 支持通用应用层协议探测, 覆盖范围大于 WhatWeb 和 Wapplyzer; 而 WhatWeb 和 Wapplyzer 对 Web 服务识别的精细度, 超过了通用的探测器 Nmap 和 Xprobe。

专有协议包括了工业控制系统协议(Modbus、Siemens S7 和 BACnet 等等), 如表格 1 所示。Onvif 是监控设备的专有协议, 利用不同厂商的监控设备在应答报文上的差异性来识别设备的类型信息。工业控制设备常常运行一些专有的协议进行远程数据读取和控制(Supervisory Control And Data Acquisition, SCADA), 例如, Modbus 协议(即工业领域通信协议)常用于工业电子设备之间的连接, BACnet 协议用于楼宇自动化控制, 供暖、通风、空调、灯光控制和门禁控制等等。研究人员 Z. Durumeric[6]分析了 Modbus 和 Siemens S7 协议, 并采用主动探测和单包方式, 发现了数万个基于 Modbus 和 Siemens S7 的工业控制设

备。Claude[21]等研究人员通过网络望远镜, 分析了 15 种工业控制协议, 解析了这些协议的数据报文, 并推测当前网络空间中工业控制设备的状况。Xuan 等人[22]进一步解析了 17 种主流的工业控制协议, 并提出了工业控制快速发现算法,可以在 24 小时内完成 40 亿 IP 地址空间工业控制协议的搜索。专有协议在设备具体实现上的巨大差异, 加大了采用专有探测包进行搜索的难度。专有协议的报文格式需要在对协议本身理解的基础上, 进而有效地解析这些探测包, 提取应用层服务的相关信息。因此, 专有协议需要花费大量精力去分析哪些指令和报文可以用来识别, 目前国内外缺少对专有协议解析的研究工作。

3.3 设备种类

网络空间中存在着海量、异构的物理设备, 包括办公设备、监控设备、网络设备、工业控制设备等等。物理设备既可以作为终端节点访问网络空间的服务, 也可以作为网络空间的基础设施, 支持数据转发和传输。设备种类信息包括设备类型、厂商和品牌型号。图 3 展示了设备种类信息的分层结构, 设备种类信息可以分为: (1)设备类型, 包括监控摄像头、网络打印机、路由器和工业控制设备; (2)设备的厂商信息, 常见的有大华、海康威视、D-Link 和 Netgear; (3)设备的型号信息, 即厂商生产的产品信息。海康威视(Hikvision) 是监控摄像头的生产厂家, D-Link 是生产路由器和监控设备的厂商, 发布了上百种不同型号的产品, 包括 DXS-3400 系列的交换机, DCS-470X 系列的摄像头等。识别设备种类信息, 即识别该设备属于哪种类型、来自于哪个厂商和具体的产品型号。

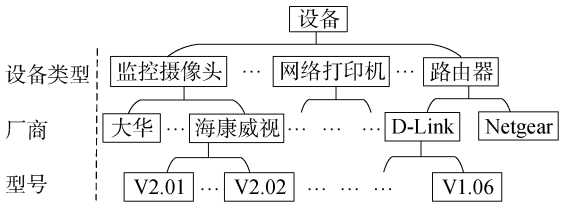

图 3 设备种类信息

Figure 3 Information of Device Type

研究人员 Cui 等人[23]提出了基于设备应用层通用协议(HTTP/SSH/TELNET 等)的差异性, 通过人工手段, 提取协议的特征字段来生成设备类型指纹。Cui 的工作[23]是第一个大规模利用设备指纹, 搜索网络空间中的物理设备信息的研究工作, 但是基于手工提取的设备种类指纹较为简单, 也无法验证识别的准确率和召回率。Nmap[3]的指纹库包含了 Cui 的研究工作所涉及到的设备种类指纹[23], 其利用正则表达式来匹配识别设备种类信息。现阶段的设备种类信息的识别, 都是利用通用协议关键字段的差异性来识别的。Xuan [9]在 2018 年的 USENIX 上, 结合利用自然语言处理技术和标语抓取技术, 自动生成了设备种类的规则。相对于 Nmap 识别指纹库, Xuan 的工作不需要训练数据, 在一个星期之内, 生产了 10 万条设备种类的规则, 并且规则能识别更细粒度的设备种类(图 3 所示)。

研究人员 A.Khakpour [24]提出了利用 TCP 数据包的特征来识别防火墙类别。A.Khakpour 提取防火墙进出口的数据流量和 TCP 字段的特征值, 结合机器学习算法训练模型, 识别防火墙种类信息。 S. Radhakrishnan[25] 发现了设备硬件时钟偏差导致的网络行为差异, 提出基于神经网络的指纹生成算法 GTID, 通过实验证明了该算法不仅可以发现移动设备(iPads, iPhones, Google Phones)和协议类型(Skype, SCP, ICMP), 而且可以发现未定义的新设备, 在发现虚假设备方面也有一定表现。

工业控制设备种类信息的识别, 是基于设备运行工业控制协议的前提下进行的。研究人员 Claude [21]和 Xuan[22]等人解析了工业协议, 在全球范围内找到了二十多万种工业控制设备, 基于工业控制协议信息, 分析了这些物理设备的基本功能、所属类别以及设备在空间和时间的分布规律。文献[6]采用纵包(Crowdsourcing)的方式, 鼓励研究人员和开放人员利用插件脚本的形式, 增加设备种类识别的指纹。识别方式和 Nmap[3]工具相同, 采用正则表达式进行匹配, 目前支持的设备类型较少且识别能力有限。

设备厂商会将监控设备种类信息硬编码在设备应用层服务。研究人员 Qiang[26]发现不同厂商的监控设备具备不同的图形化交互界面(Graphic User Interface, GUI)。研究工作[26]提出了基于 GUI 的监控设备种类信息识别算法, 通过 HTTP 协议自动提取监控设备的 web 页面信息, 并在整个 IPv4 空间中发现了 160 万个监控设备。北美欧洲大规模断网的 Mirai 病毒干扰的物联网设备大多都属于监控设备, 例如摄像头、NVR 和 DVR。研究人员 M.Antonakakis[2]利用 HTTP 的标语信息存储的厂商硬编码信息, 并结合 Nmap[3]的指纹库, 分析了这些监控设备的厂商和品牌种类信息。

目前, 设备类型信息的搜索采用基于分类模型的指纹生成技术[26], 或者采用基于标语的指纹生成技术[2-3, 23], 在四章节, 本文将会介绍识别的关键技术。

3.4 标识信息

设备种类信息代表了同一类别的设备信息, 而

设备标识表示单个设备的信息。标识信息即 ID, 是区别设备之间的标识符。典型的标识方法包括: 基于IP地址的标识, 基于MAC地址的标识和基于Cookie的标识。

IP 地址是由组织机构 IANA 进行统一分配和发布的, 同时, 动态地址分配协议(Dynamic Host Configuration Protocol, DHCP)使得每个 IP 地址具有一定租赁期限。超过租赁期限后, 物联网设备的 IP 地址会发生改变。研究工作[28]通过大规模测量和分析发现, IP 地址的租赁期限从一个小时到数个星期不等, 平均租赁期限约为 24 小时。动态变化的 IP 地址无法作为标识信息。MAC 地址是 48 比特(6 个字节), 其中后面 24 比特是由厂商生成和定义的。MAC 地址只能在局域网空间内获得, 可以唯一标识单个设备。网络空间中, 我们无法获得 MAC 地址信息。Cookie 或者用户登入信息, 要求应用服务器采集或者用户主动输入, 应用范围非常受限。网络空间中, 这些标识的搜索具有很大的局限性, 无法作为物联网标识信息。本章节介绍了基于设备本身物理信息标识的相关研究工作。由于设备的硬件和系统在设计和实现上具有差异性, 因此可以通过间接的方式, 将获得的物理特性作为设备的标识信息。

研究人员 T.Kohno[13]提出将时钟偏移的物理信息作为设备的标识。网络时钟协议(Network Time Protocol)是计算机系统通过发送数据包来校正系统时钟的网络协议, 不同的设备运行时, 硬件的实现和系统运行也有所不同, 这就造成了设备的时钟和标准时钟的同步出现了细微的差别。T.Kohno 的研究工作提出了结合主动探测和被动侦听两种方式, 提取一段连续 TCP 报文头部的时钟值, 其中 t_d 表示被探测对象的时钟值, t_s 表示探测服务器的时钟值。时钟的偏移值分别为: $x_i = t_d^i - t_d^1$ 和 $y_i = t_s^i - t_s^1$。基于连续的的偏移对 (x_i, y_i), 通过多项式拟合的方式, 即 $y = a_0 + a_1 x_1 + a_2 x^2 + \cdots + a_k x^k$, 将得到的参数 $(a_0, a_1, \cdots, a_k)$ 来表示设备的物理特性。这种方法的优点在于可以通过探测包和响应包之间的时间戳差异建立设备的物理指纹, 这种指纹和具体设备相关, 难以修改, 因而可以作为设备的标识信息。

物理信息的限制在于两个方面: (1)网络抖动、丢包和时间差异值不稳定的问题, 会导致物理特性不稳定和误差值的提高。S.Zander[29]在此基础上, 提出时钟偏移的测量主要有两方面的误差: 网络干扰和时间戳量化误差。时钟量化误差产生的噪声比网络干扰大一个到多个数量级。S.Zander 的研究工作提出了一种基于时钟同步抽样的测量方式, 很大程度地减少了时间戳量化的误差。Xuan [30]提出基于分层结构的设备时钟偏移, 首先采取额外的信息对网络空间中的物理设备进行分层, 包括设备种类、所属机构、和位置信息, 在分层结构的末端节点, 再采用时钟偏移计算物联网设备的物理特性, 以此减少网络抖动带来的影响。(2)计算设备的时钟偏移需要大量连续的数据报, 需要花费很高的代价, 不利于大规模网络空间的探测。获取设备的时钟偏移值, 需要连续的 2000 到 3000 个数据包, 否则会带来很大的误差。

物联网标识信息还可以用来发现和检测网络空间的异常行为。研究人员 D.Formby [31]发现真实的和伪造的工业控制设备的响应包在时间上具有差异性。D.Formby 的研究工作利用响应时间的差异提取设备的标识和模型, 如果发现伪造的设备, 那么认为当前的工业控制系统被伪造或被入侵导致物理特性改变, 从而保护工业控制系统。研究人员 D.Urbina[32]认为工业控制设备的电压在时间序列上的变化规律可以作为标识信息, 即使隐藏攻击行为, 也会导致正常的设备标识发生改变。D.Urbina 的研究工作利用物理特性提取设备标识, 建立正常设备模型, 有效地发现工业控制系统中的攻击行为和异常现象。

4 物联网搜索的关键技术

网络空间中的物联网信息具有三个特性: 海量动态、异构和自组织。数以亿万的物联网设备接入网络空间, 形成了海量的物联网信息。这些物联网信息无时不刻在变化, 例如动态 IP 地址, 新设备的安装和旧设备的拆除。设备的移动性进一步加剧了网络空间物联网信息的动态性。海量动态的物联网信息的探测具有实时性要求。物联网信息各不相同, 数据格式异构, 设备种类繁多, 识别技术需要精细化地获取物联网信息。物联网应用在网络空间是自组织式的分布, 探测技术需要能够遍历整个网络空间, 识别技术需要全面地发现和识别物联网信息。本章节主要从探测技术和识别技术分别介绍相关研究工作。

4.1 探测技术

探测技术已经有了近 20 年的发展历史, 图 4 描述了物联网信息探测技术的框架。物联网信息包括传输层信息, 网络层信息和应用层信息; 根据物联网信息收集方式的不同, 探测技术可以分为主动探测和被动侦听; 根据探测器、探测数目和探测顺序的

不同, 物联网探测可以采取调度策略完成网络空间探测。下面分别介绍探测技术各个模块的相关工作。

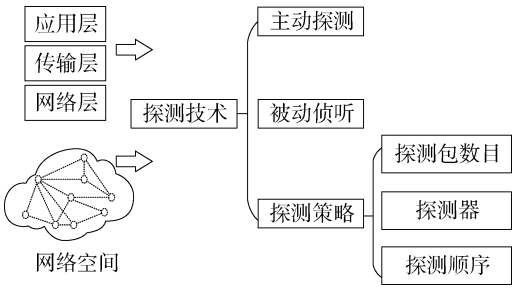

图 4 网络空间中物联网探测技术

Figure 4 IOT Detection Technology in Cyberspace

4.1.1 主动探测

主动探测是指探测器向网络空间发送探测包并接受响应包, 通过分析和挖掘响应包来推测被探测对象的相关信息。攻击者在发起一次攻击之前, 会使用主动探测来发现脆弱设备和信息; 安全研究人员利用主动探测检测当前网络环境的基本情况, 及时发现高危漏洞, 更新安全补丁。主动探测针对不同的物联网信息会发送不同的探测报文。

在网络层, 探测器会发送 ICMP 数据包到一个目的 IP 地址, 如果得到响应, 就可以得知存活在该 IP 地址上的当前主机或设备。ICMP 不需要指定特定的端口, 主动探测可以通过是否应答来得知被探测对象是否存活。早期的探测技术[7]都是采用 ICMP 数据包完成网络空间的搜索任务。常用的工具 Nmap[3]和 Xprobe[14]都支持 ICMP 数据包探测。目前, 许多防火墙和入侵检测系统都会阻塞探测器的 ICMP 数据包。

在传输层, 主动探测通过发送 TCP 或者 UDP 数据包来发现被探测对象开放了哪些端口和服务。TCP 探测包需要建立状态连接, UDP 探测包则不需要。如果被探测对象返回相应的数据包, 探测器就可以得知开放的 TCP 端口和 UDP 端口。主动探测基于 TCP 链接方式, 可以分为半链接和全链接。半链接探测是指探测服务器发送一个 TCP-SYN 探测报到被探测对象的某个端口(例如 21 端口), 如果被探测对象开放了 FTP 应用服务, 就会返回一个 TCP-ACK 响应包。快速扫描工具 Zmap[8]采用这种半 TCP 链接的方法实现了全网探测。研究人员[33]也提出了基于这种半链接方式探测网络的拓扑结构, 能够在一个小时就可以找到 IPv4 空间内所有路由信息(即/24 网段信息)。采用全链接方式, 探测服务器和被探测对象则会建立三次握手协议, 进而探测应用层的数据, 直到探测服务器不再响应, 此次探测任务结束。这种全链接的探测方式有 2 个限制: (1)TCP 链接会消耗被探测对象的资源; (2)全链接方式的探测时间迟较长, 无法快速完成大规模物联网信息搜索。

在应用层, 主动探测通过发送应用层的探测包来解析相应的应用服务, 章节 3.2 阐述了应用层服务搜索的相应研究工作。应用程序信息是指运行在被探测对象上的应用和服务的相关信息。研究人员 Genevieve[34]结合了主动探测和被动侦听来量化网络空间的应用服务信息。Nessus [35] 利用发送到网络空间的探测包来评估了脆弱性的主动探测工具。中科院信工所研究团队通过主动探测技术发现网络空间的监控设备[26]和工业控制设备[22], 并做了进一步的监测和分析。

主动探测的优点在于, 它可以灵活地选择探测范围和需要探测的内容。通过布置多个探测器, 主动探测可以实时地获取被探测对象的信息。不足之处在于容易被防火墙阻塞, 不能够穿透网关和 NAT(网络地址转化), 甚至可能会影响远程主机的正常运行。如果发送数据包速度过快, 很容易被认为是一种攻击行为, 大量的探测包会对网络环境造成不良影响。因此, 主动探测需要尽可能地减轻对网络空间的影响, 保证合法性和合理性, 这是网络空间中物联网搜索需要铭记的地方。

4.1.2 被动侦听

被动侦听通过布置在网络边界的探测器, 收集网络空间的数据, 分析和挖掘数据包进而推测被探测对象的相关信息。探测器可以通过专用硬件或软件, 结合数据包解析技术获取相关的物联网信息。攻击者常常采用被动侦听来发现当前网络空间的脆弱点, 为进一步的攻击做准备; 入侵检测系统(IDS)则利用被动侦听技术发现网络空间中的异常和安全隐患。

被动侦听主要捕获传输层和应用层的服务信息, 它可以在路由器上复制一个虚拟端口, 将所有经过路由器的数据包都复制到探测器上, 因而不影响当前路由或者网关的正常运行。在传输层, 被动侦听主要解析 TCP 和 UDP 数据流。基于 TCP 数据包, 被动侦听需要捕获 TCP 连接建立的消息(即 SYN 数据包), 得知三次握手完成的情况, 若 TCP 数据包的(SYN/ACK)消息成对存在, 则 TCP 服务建立成功。基于 UDP 数据包, 由于 UDP 不要求被探测对象返回响应包, 所以被动侦听需要通过观察流量来识别 UDP 服务情况。研究人员 F. Donelson[36]通过研究 TCP 的头部信息分析和推测了网络空间中的 Web 服务。在应用层, 被动侦听技术只能分析 HTTP, FTP 这样的主流应用层协议。针对通用协议的数据包, 报文字段较

为固定, 因此很容易推断出被探测对象的信息; 针对专有协议或者加密的应用层数据包, 被动侦听则通过流量来推测具体的应用服务情况。

被动侦听技术常常用来检测当前网络环境中的资产、异常和设备信息, 结合 TCP, ARP 和 ICMP 协议来生成应用程序指纹, 分析检测出的数据包, 从而实现匹配。Wireshark 是常见的被动侦听分析工具, 它可以直接解析 IP 包、TCP 包、UDP 包等报文。P0f[15] 利用被动侦听数据包来识别出当前网络环境的操作系统信息。入侵检测系统 Bro 和 Snort 都是利用被动侦听技术来收集网络数据包, 从而发现异常和入侵行为。研究人员 Michael Bailey[37]提出了被动侦听设施的分布式检测算法, 用来分析网络空间的潜在危险, 并且利用它们进行局域网内的入侵检测和安全评估。全球规模最大的被动侦听设施是 CAIDA 的网络望远镜(Telescope)[38], 256 分之一的 IPv4 地址空间(即 156 万个 IP 地址)都可以被网络望远镜实时侦听和分析。许多研究工作都是基于网络望远镜来探测和发现网络空间中的安全问题。David Moore 等研究人员[39]指出了如何利用网络望远镜去检测网络空间中的安全事件, 诸如分布式拒绝服务攻击, 僵尸网络, 蠕虫和木马的传播。类似于网络望远镜, 瑞士理工的研究人员 Eduard[40]通过布置 46 所校园和研究机构的被动侦听设施, 建立分类模型来预测网络空间的主动探测行为和网络中不可达的区域, 总结了基于被动侦听识别的网络空间中潜在的安全事件。

被动侦听最大的优点在于对被探测对象的影响可以忽略不计, 它不需要给被探测对象发送任何数据包, 不消耗资源。同时被动侦听也不会被认为是一种攻击行为, 并且还可以捕获那些被防火墙阻塞的数据包, 具有很强的穿透性。被动侦听最大的局限在于其缺乏灵活性, 无法应用于大规模的网络空间探测。如果这段网络空间的数据包不经过探测器, 那么被动侦听无法得到任何信息。

研究人员 J.Richard[10]总结了主动探测和被动侦听的优缺点, 主动探测不需要布置监测服务器, 但是会消耗被探测对象资源, 且发现不了被阻塞的设备和服务; 被动侦听的使用范围受到监测服务器布置范围的限制, 但却不会消耗被探测对象资源, 并且可以发现一些间歇性的和被阻塞的设备和服务。

4.1.3 探测策略

根据探测服务器、探测包的数目和探测顺序的不同, 网络空间搜索可以设计不同的策略和算法去发现和识别物联网设备。本章节总结已有的相关工作, 并阐述这些因素对网络空间探测策略的影响。

探测器。当探测器数目较少时, 探测方式较为简单, 通过发送数据包并接受和解析响应包来完成物联网搜索的任务。少量的探测器会受到许多限制, 比如上下行带宽会导致探测时间过长, 数据失真使得可信度下降等等。当探测器数目较多时, 可以分配探测任务到多个探测器上, 同时设计调度算法分布式地执行探测任务, 提高探测速率。C. Gates[41] 提出了分布式的、多源服务器协作探测网络空间的方式, 从而节省了探测时间。这种分布式搜索方式很难被察觉, 探测任务调度问题转化为集合覆盖问题, 在理论上优化了搜索服务器数目和位置固定的分布, 通过分析得出多个探测器之间存在时间和空间上的关联关系。

服务器的位置也是影响网络空间探测的重要因素。在内部网络空间中, 内部网络系统管理员利用探测器发现可疑的设备和安全隐患, 及时打上补丁来保证内部网络安全, 进行安全审计工作。研究人员 D.Whyte [42]研究发现, 内部网络空间搜索是保证内部网络空间安全的重要技术, 他利用内部网络搜索结合 DNS 异常检测, 发现了企业网络中被蠕虫感染的计算机和嵌入式设备。在外部网络空间中, 探测器需要考虑到搜索行为对外部网络行为所造成的影响, 否则会被认为是一种网络入侵行为, 带来法律问题。探测器探索网络空间时, 需要考虑到网络地址转换(Network Address Transformation, NAT)造成的影响, 网络地址转换通过将内部网络保留的 IP 地址和外部网络空间全球唯一的 IP 地址进行转换, 解决了 IP 地址不足的问题并隐藏了内部网络的信息。

探测顺序。国际组织机构 IANA 将 IPv4 地址空间连续分配给不同的国家和组织机构, 互联网 IPv4 地址空间大约有 $n=2^{32}$ 个数目, 约等于 40 亿。我们用集合 $S=\{d_1,d_2,\cdots,d_n\}$ 表示探测网络空间, 其中 d_i 表示空间中的具体地址, 探测空间大小|S|等于 n。针对网络空间 S 有三个探测顺序: 顺序探测, 块探测和随机探测。顺序探测, 即不打乱集合 S 中地址的次序, 按照 $d_1,d_2,\cdots,d_n$ 的顺序将网络空间中的每一个地址都探测一遍, 保证了探测的完整性。这种探测方式很简单而且容易实现, 很多网络测量的研究工作都采用这种顺序的探测策略。顺序探测最大缺点是探测报文都是连续发送给同一地址空间, 消耗网络资源的同时也容易被认为是一种入侵行为。块探测首先将集合 S 中的地址空间划分为数个地址块, 每一个地址块的大小相同, 即 $S=\{s_1,s_2,\cdots,s_k\}$, 其中 $S_i=\{d_i^1,d_i^2,\cdots,d_i^{ni}\}, n=\sum_{i=1}^{k} n_i$。探测顺序是分别抽

取每个子地址空间块 S_i 的地址进行探测, 直到探测完成。Heidemann[43]等研究人员采用块探测完成了 3.4 亿个 IPv4 地址空间探测的任务。随机探测是指打乱集合 S 中地址空间的次序, 然后完成探测任务。许多相关工作都采用了随机探测顺序地址随机化的探测方法, 一般采用置换群的线性同余随机化方法, 保障了地址空间探测的完整性和每个地址仅探测一次的准确性。研究人员 Leonard [44]分析了随机探测顺序对被探测对象的影响, 并认为一般的入侵检测系统和防火墙很难发现这种随机探测方式。Z. Durumeric [45]提出的 Zmap 工具, 就是采用随机探测顺序完成了网络空间探测任务。

探测包数目。探测器发送探测包, 是收集被探测对象数据和识别信息的前提。被动侦听通过布置探测服务器直接采集数据包, 探测包的数目为 0, 即无需要发送任何探测包, 只有主动探测才需要发送探测包, 进行物联网信息收集。

当探测包的数目为 1 时, 探测方式可以称之为水平扫描。研究人员 Leonard [46]利用水平扫描, 探测了六种网络协议(DNS, HTTP, SMTP, EPMAP, ICMP 和 UDP), 在 3.5Gbps 的网络带宽下, 24 小时内完成了 IPv4 的网络空间探测。目前最流行的网络探测工作 Zmap 和 MASSCAN, 就是将一个 TCP-SYN 数据报文随机地发送给每一个被探测地址, 从而完成网络空间的存活性测量。理论上, Zmap 和 MASSCAN 在拥有千兆带宽的条件下, 可以在数十分钟内完成 40 亿个 IPv4 地址空间的探测, 但实际过程中受到网络带宽和丢包的影响, 过快的探测速率会导致探测覆盖率大大降低。中科院信息工程研究所研究人员发现[22], 当探测速率超过每秒 50 万个探测包时, 很多被探测对象不会返回数据。

当探测包数目很多时, 探测方式可以称之为垂直扫描。我们可以发送多个数据包对单个设备进行全方位的信息收集, 包括开放端口、正在运行的服务, 操作系统类型和版本、设备种类、厂商和型号、固件版本等等。然而, 研究人员认为发送过多的数据包是一种危害网络空间正常服务的行为。垂直扫描收集的信息虽然比较全面和丰富, 但其需要建立完整的 TCP 连接且等待回复的时间较长, 效率较低。

4.2 识别技术

网络空间中识别物联网信息需要提取其相应的指纹, 图 5 描述了基于指纹的物联网信息的识别过程, 通过指纹生成技术将网络探测的不同数据信息转换为指纹, 进一步匹配得到相应的物联网信息。物联网信息的指纹需要三个条件: (1)输入: 探测包和响应包; (2)处理函数: 匹配规则或者代价函数; (3)输出: 物联网信息。指纹生成技术可以分为两类: 基于分类模型的指纹生成和基于标语的指纹生成, 本章节分别从这两个方面阐述物联网信息指纹生成技术的相关研究工作。

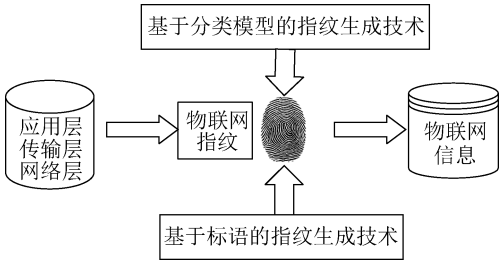

图 5 基于指纹技术的物联网信息识别

Figure 5 Internet of Things Information Recognition Base on Fingerprint

4.2.1 基于分类模型的指纹生成技术

基于分类模型的指纹生成技术需要将物联网信息的训练数据进行特征值选取和学习算法的分类, 本章节从特征值选取和学习算法两个方面来阐述指纹生成技术的相关研究工作。

(1) 特征提取

特征提取是基于网络空间探测收集的数据, 选取具有区分度的特征向量。如果采用主动探测, 训练数据是成对<探测包, 响应包>出现的, 如果采用被动侦听, 训练数据的形式是<空, 响应包>, 其中数据包可以来自于传输层、网络层和应用层的协议。

传输层数据的报文头部可以用来提取相关特征值。TCP 数据报文头部包括了协议序号初始值(ISN)、IP 生存时间初始值(TTL)、TCP 滑动窗口尺寸、最大分段长度等信息, Nmap [3] 工具提取了 6 个 TCP 包和 8 个 UDP 包报文头部的信息, 将此作为识别操作系统版本的特征值。探测器和被探测对象可以根据网络时钟同步协议, 将记录的当前时钟作为识别设备的特征值。T.Kohno[13]利用 TCP 数据报文头部字段存储的时间戳来计算被探测对象的时钟偏移, 生成设备标识符的特征值。研究工作[11]则利用数据包重传超时时延作为识别操作系统版本的特征值。传输过程中数据包重传超时时延方记录了收到数据包的超时上限值, 若未收到确认数据包, 发送方会重传这个(Retransmission timeouts, RTO)数据包, 若收到了数据包, 则重置重传定时器。探测服务器发送 TCP-SYN 数据报到被探测对象, 被探测对象会将返回的 ACK-SYNs 数据报重传的个数和时延包含在 TCP 报文头部的 TTL 中, 当设备运行不同版本的操

作系统时, 所包含的 ACK-SYNs 的时延和重置次数是不同的, 因此这种差异性也可以作为操作系统版本的特征值。数据报来回时间延迟(Round-Trip Time)可以记录探测服务器和被探测对象之间数据报的延迟, 因此利用在 ICMP 或者 TCP 数据报文头部设置的存活时间(TTL)的数据, 结合数据报来回时间的延迟, 可以探测当前网络空间的拓扑结构。研究人员 Robert [33] 利用半链接的 TCP 探测和来回时间延迟, 在一个小时内完成了 IPv4 所有网段的拓扑结构的探测。Caballero [12]提出了一种基于比特流的特征值提取方法, 将每对探测报和响应包作为候选集合, 查询并提取有效的、区分较大的比特流段, 以此作为设备识别的特征值。这种方法需要预先了解协议内容, 否则无法获得有效的比特流段。流量信息也可以作为生成指纹的特征值, 研究人员[24]发送了四组报文(完全相同的 TCP 包, 不同源端口的 TCP 包, 完全相同的 UDP 包和不同源端口的 UDP 包)到防火墙, 将防火墙处理每组报文的时间差异作为指纹特征。

(2) 学习算法

选取好特征值和训练数据后, 物联网信息的识别还需要利用学习算法训练和推导出一个代价函数, 输出当前设备的信息。

学习算法可以分为监督式学习和非监督式学习。监督式学习, 要求训练数据带有标签, 而非监督式学习则不需要。文献[12]利用决策树学习算法, 通过数据报比特流的特征值获得的数个决策树模型作为代价函数, 并作为识别设备操作系统版本的指纹。研究人员 David[18]通过实验发现基于分类模型的指纹存在几个问题, 它不仅需要大量带有标签的数据, 而且只在识别小规模的设备种类时性能较好。文献[47]提出了半监督的聚类算法, 训练数据中包括了带有标记的数据包和未带标记的数据包, 将网络协议特征进行分簇, 从而将网络信息划分到不同群组, 然而此方法依然只能适应于小范围的设备操作系统信息的识别。研究人员[48]在前者工作的基础上, 引入了两种计算指纹特征距离的算法, 该工作采用 ROCK 和 QROCK 两类算法提取指纹特征值, 结合监督式的支持向量机学习算法和非监督式的分簇算法生成操作系统指纹信息。研究人员 Qiang[26]采用 SVM、朴素贝叶斯、决策树和神经网络等监督式学习算法, 生成监控设备的指纹, 利用 HTTP 流量和报文在结构上的相似性识别了网络空间中的监控设备。在监控设备识别方面, Qiang 的研究工作具有比较高效和正确的分类性能。

4.2.2 基于标语识别的指纹生成技术

基于标语的指纹生成技术, 即通过物联网信息的训练数据进行标语信息的选取和规则的匹配。物联网设备的厂商开发、生产和发布设备产品之后, 会在网络协议栈中嵌入信息, 我们将这些信息称为标语信息。我们可以通过分析被探测对象的数据报文头部或者协议本身来找到标语信息。基于标语的指纹生成技术包括三个步骤: (1)收集网络空间探测数据; (2)提取和存储标语信息; (3)匹配探测数据和标语。如果能够分析数据报文的格式, 那么我们可以从中提取相关的标语信息; 如果数据报格式未知, 则需要逆向工程或者协议分析来获取标语信息。

文献[23]是第一个利用标语信息进行设备指纹提取的研究工作, 基于人工采集的方式, 提取应用层协议的特征字段, 作为嵌入设备的指纹。研究人员 Caballero. J [50]提出了 Polyglot, 利用动态二进制代码分析完成了协议逆向分析, 提取数据报文的特殊字段, 包括方向字段、字段分隔符和协议关键字等, 作为设备指纹。然而, 基于协议逆向的 Polyglot 对于提取长度字段等信息具有较低的正确率, 不能在实际工作中自动生成设备指纹。Wondracek[51] 提出了利用动态污点分析技术动态提取应用程序的执行过程, 并且分析程序如何处理协议消息, 实现动态的协议逆向。该方法能够正确的提取协议格式以及长度等字段, 在分析之前需要手动将消息进行分类, 进而自动生成设备指纹。基于协议逆向和污点分析的标语生成指纹, 要求大量的人力和时间, 不适合大规模设备指纹的自动生成。研究人员 D. Brumley [52]则从白盒测试角度提出了用符号执行方法提取指纹模型的技术, 通过分析不同系统上的协议固件, 找到不同系统实现协议的差异性, 进而推断能够触发差异的有效探测报文。这种方法需要对程序进行修改和插桩, 时间复杂度较高, 且需要大量的人工分析。研究人员 P. Comparetti [53]提出了 Prospex 工具, 实现自动地推断协议状态, 其不仅依赖于消息的结构特征, 而且将每个消息对服务器行为的影响进行分类, 自动将消息分成不同的类型。

研究人员 Z.Xu[49]发现当设备被恶意程序(C&C 方式)感染后, 设备返回的 HTTP 报文头部会将自身的 IP 地址和服务信息编码成难以理解的字段。研究工作[49]建立了有效的、高效的且能够用于主动检测远程恶意服务器的指纹, 利用包括污点追踪, 动态切片和符号执行等动态二进制分析技术, 解决了当前探测方法在用于大量恶意软件的基于 C&C 协议情况下的不适用。Mirai 病毒感染的监控设备识别[2]也

是采用基于标语的方式生成的设备指纹。

5 物联网信息搜索的应用

本章节主要从互联网空间测量和大规模安全事件分析这两方面来介绍网络空间物联网搜索的典型应用。

5.1 互联网空间测量

互联网空间测量是指探测IPv4空间的可见(可访问)地址, 提取相关信息, 作为大规模物联网信息搜索的典型应用。网络空间探测要大于互联网空间测量, 许多物联网信息都隐藏在网关、路由之后, 例如企业网或者家庭网络。互联网测量空间具有 40 亿个 IPv4 地址, 利用探测技术收集可见地址的相关数据, 并结合指纹技术识别物联网信息。

互联网测量早期的工作主要集中于如何提高网络测量的速率。J. Heidemann [43]提出了加快探测速度的三个关键因素: 并发, 分段, 探测包数目, 并以此提出了一个优化探测算法, 通过降低探测链接时间、去除重传等手段, 在 30 天内完成了整个 IPv4 空间的探测。此外, J. Heidemann 的研究工作 Census 为了加快全球网络空间探索速率, 采用普查方式探测整个网络空间和抽样方式推测其余部分网络空间。Leonard [54]提出了 IRLscanner, 一种通过地址排列和拆分算法来调节扫描速率的方法(GIW, Globally IP Wide), 结合并行处理服务器, 只要 24 小时就能完成 IPv4 空间的探测。IRLscanner 不仅降低了被入侵检测设备发现的概率, 而且大大减少了原来花费数周甚至数月的扫描时间代价, 搜索过程变得很“友善”, 不会影响被探测网络的正常使用。研究工作 [55]介绍了一种利用僵尸设备作为搜索服务器进行分布式搜索以完成全球网络空间搜索任务的方法。美国密歇根大学的 Z. Durumeric [8]提出了一种快速的主机存活扫描器—Zmap, 与之前研究工作的不同点在于 Zmap 不是真正建立 TCP 链接, 而是进行半链接的无状态搜索, 它可以在单台计算机上数个小时内完成整个 IPv4 地址空间的单端口扫描, 且两轮达到 98%以上的主机覆盖率。如果充分利用当前网络带宽, Zmap 可以在 1Gbps 上行带宽下 45 分钟内完成全网的探测。更进一步, 如果网络带宽达到 10Gbps 上行带宽, 可以在理论值 5 分钟内完成全网空间的探测。对于 ZMap 而言, 无论是 45 分钟还是 5 分钟, 这种探测都是在网络上下行带宽充分、无阻塞的前提下完成的。然而实际情况常常并不满足前提, 当速率过快时, 设备发现率会大幅度降低。因此, 密西根大学的研究团队搭建了网络空间搜索引擎——Censys [6], 并公开了物联网设备搜索引擎的实现细节、关键源码和数据。

近几年, 许多研究工作基于网络测量收集的数据分析了具体的物联网信息。Krishnamurthy. B [56]. 提出了基于网络空间上下文情景来发现运行 Web 客户端的设备, 考虑到边界网关协议(BGP)会聚合大量的 Web 客户端, 相同边界的设备会在相似管理的控制中, 他利用分簇算法将其划分为不同集群, 进而加快运行 Web 客户端的设备发现速度。Cai. X [57]通过主动探测探测了百分之一的网络空间, 并设计聚类算法估计了网络空间的使用情况。Cai. X 考虑到邻近的地址一般具有相似的特性, 因此利用地址块进行聚类分析推断其他网络空间的情况, 包括地址使用率、动态地址分配的程度、变化以及网络区域管理策略等。Hong.Y [58]提出 PIPMiner, 自动分析流行的 IP 地址 PIPs(Populated IP Addresses)。网络空间中某些地址会提供资源和服务, 大量用户会请求和访问它们, 因此 PIPMiner 结合机器学习和时间序列分析技术来识别这些特别的网络空间地址。Xu.K[59]探索了在同一个网络前缀的互联网中终端主机行为的相似性, 利用二分图对网络流量建立模型, 并建立单模式映射图(one-mode projection graphs)采集终端主机流量行为的相似特征, 通过应用一个简单高效的谱聚类算法将相同地址前缀的终端主机分成了不同的行为集群。Quan.L [60]提出了 Trinocular, 一个利用主动探测来了解边缘网络可靠性的运行中断(outage)检测系统, 其利用简单的以故障为中心的网络空间模型, 结合贝叶斯推断驱动的探测方式来推断出当前的网络空间状态。该方法在目标网络中仅产生 0.7%的负担, 并且能够在 330 秒之内检测出所有至少持续 11 分钟故障的网络空间。D. Springall 等人[61]结合 Zmap 和应用层协议 FTP 的标语, 在 40 亿的 IP 地址空间中找到了 1300 万个 FTP 服务器, 其中 110 万个允许“匿名”(公共)访问, 并通过分析发现 20,000 台 FTP 服务器允许公共写入访问, 这为全球网络空间安全带来了很大的隐患。Xuan 等人[23]分析了 17 种主流的工业控制协议, 提出了工业控制快速发现算法, 在全球范围内找到了二十多万个工业控制设备, 并分析了设备在空间和时间的分布规律。Qiang 等人[26]发现监控设备的 web 页面可以作为识别监控设备的指纹, 因此提出了利用 HTTP 协议提取监控设备的 web 页面信息, 结合机器学习算法生成监控设备的指纹。Claude et.al 等研究人员[21]通过网络望远镜分析了十五种工业控制协议, 解析了这些协议的数据报文, 并推测当前网络空间中工业

控制设备的现状。

5.2 大规模安全事件分析

随着物联网在各个领域的不断发展, 近年来针对物联网相关的安全事件层出不穷, 网络空间物联网信息搜索则可以帮助研究人员大规模地分析这些安全事件。我们通过分析三类大规模安全事件, 来说明物联网信息搜索对网络空间安全的作用。

僵尸网络是一种网络攻击手段, 许多设备感染了蠕虫、病毒或者木马就会成为僵尸网络的一部分, 并对正常运行的服务进行攻击, 造成严重的经济损失, 带来国家安全隐患。早期, S.Staniford 研究人员[62]分析了三种类型的蠕虫病毒, 分别是 Code Red I, Code Red II, 和 Nimda, 并发现将近超过 1000 万台主机成为了僵尸网络。Staniford 认为, 成为僵尸网络的设备还会利用剩余的时间和计算资源进行网络空间探测, 发现更多潜在的设备, 从而进行下一步的感染。Alberto.D[64]利用来自于 UCSD 网络望远镜的数据发现了之前没有公开过的大规模利用 300 万 IP 地址僵尸网络进行的隐蔽扫描行为 —— Sality 僵尸网络对 SIP 服务器的水平扫描, 并对该扫描行为进行了测量和分析。Alberto. D 发现为期 12 天的扫描发源于大约 3 百万个不同的 IP 地址, 这些僵尸网络使用了分布式协作的扫描策略, 针对 VoIP 相关的 SIP 服务器进行攻击。近期 Mirai 病毒感染的物联网设备[2]主要包括监控摄像头、网络视频记录服务器和数字视频记录服务器, 将近 100 多万个监控设备形成了僵尸网络, 并对 DNS 服务器进行了 DDoS 攻击, 造成了大规模的网络中断。

物联网设备存在大量的安全漏洞和隐患, 因此对物理设备的安全分析是保障网络空间安全的前提。A. Cui 等人[65]发现惠普打印机升级固件时, 恶意用户可以篡改或者攻击网络传输数据, 造成打印机功能损坏。他们分析了整个 IPv4 空间内所有可以访问的打印机, 发现 9 万个惠普的网络打印机可以被攻击, 而且这些打印机大多安置在政府、教育机构和其他敏感环境。2014 年 4 月, 心脏出血漏洞(Heartbleed)让互联网倍感意外, 这个漏洞是商业互联网出现以来最重要的漏洞之一, 攻击者可以利用这个漏洞对服务器站点的内容进行远程读取并获取敏感信息。Z. Durumeric 等人[66] 进行了全面的、基于测量的漏洞影响分析, 追踪漏洞主机的数量, 监控了随时间的发展漏洞的修补情况, 并评估和分析了其对 HTTPS 生态环境的影响以及一些企图利用该漏洞的攻击行为, 而且他们发现仍然有大约 24%~55%的服务器站点还存在着这个漏洞。Simurgh.A[67]通过全球网络空间搜索的数据挖掘了 2013 年伊朗大选期间的网络测量, 来调查伊朗的网络审查状态, 并提出了基于 HTTP 主机的拦截、关键字过滤、DNS 劫持和基于协议的限制等技术机制判断出审查基础设施的网络拓扑, 且发现该审查网络很大程度上依赖于集中式设备。

恶意服务器是网络空间安全的主要威胁, 在打击网络犯罪分子的斗争中, 准确地识别潜在的恶意服务器基础设施至关重要。Xu 等人[49]针对恶意服务器自动建立了有效的、高效的、并且能够用于主动检测远程恶意服务器的指纹, 利用包括污点追踪、动态切片和符号执行等动态二进制分析技术, 解决了当前的探测方法在用于大量恶意软件的基于 C&C 协议情况下的不适用。针对物联网设备的多样性, Z. Xu[68]通过污点跟踪、动态切片和符号执行技术对恶意二进制代码进行了动态分析, 提取恶意代码对特定请求数据包的特定处理方式, 构建恶意服务器指纹, 利用指纹进行主动探测发现恶意服务器。Zhang 等人[69]对 100 000 多台良性服务器、45 000 台恶意服务器和 4 万次重定向进行了深入分析, 从它们的位置、结构、角色和关系等角度确定了一系列恶意网络基础设施的不同特征, 并提出一个轻量级而有效的恶意服务检测系统。Liao 等人[70]在 CCS 2016 上针对恶意者使用云托管服务来进行恶意在线活动的问题, 提出了利用主动探测发现恶意云仓库的方法, 基于收集的已知恶意云仓库, 分析其网页的跳转模式, 提取出唯一表征恶意云仓库的集体特征, 并基于此特征构建了一个扫描程序, 在 Amazon, Google 等 150K 网站上检测到超过 600 个恶意云仓库。Krupp 等人[63]在 CCS 2016 上基于发动放大 DDoS 攻击之前会进行扫描的特性, 来检测放大 DDoS 攻击背后的基础设施, 首先对扫描器建立指纹, 在执行放大攻击的侦察过程中利用指纹将后续的攻击链接到扫描器, 然后使用基于 TTL 存活时间的三边测量技术将扫描器映射到发起攻击的实际基础设施上。最后确定了 34 个作为放大 DDoS 攻击源的网络, 准确率高达 98%。

6 结论和展望

随着物联网在网络空间的不断发展, 物联网相关信息和技术已成为网络空间中的重要部分, 物联网信息搜索也处于网络空间安全攻防体系中的核心位置。物联网信息搜索, 通过布置探测器, 采取主动或被动的探测技术, 结合探测策略, 收集网络空间中的相关数据, 基于物联网信息的指纹技术, 识别

网络空间中的物联网信息。本篇论文在技术层面, 阐述了物联网信息搜索相关研究工作的进展, 给出了网络空间物联网信息搜索的定义, 分析和总结了物联网信息搜索的方法、策略和识别的关键技术研究。本篇论文是总结网络空间物联网搜索信息研究进展的综述性文章。

参考文献

[1] Gartner Newsroom. 8.4 Billion Connected "Things" Will Be in Use in 2017, Up 31 Percent From 2016 [Online]. Available: https://gartner.com/newsroom/id/3598917

[2] Manos.Antonakakis ,Tim.A and et al, "Understanding the Mirai Botnet,"*26th Security Symposium({USENIX} Security 17)*, 2017,978-1-931971-40-9, Vancouver, BC, 1093—1110.

[3] Lyon G, "The Art of Port Scanning," Phrack Magazine, 1997, 7(52).

[4] J. Matherly(2009), "The Search Engine for Internet Connected Devices," Available: http://www.shodanhq.com/

[5] Censys,A Search Engine for Internet-Wide Scanning, https://censys.io/

[6] Durumeric Z, Adrian D, Mirian A, et al, "A Search Engine Backed by Internet-Wide Scanning," *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security. ACM*, 2015: 542-553.

[7] Allman M, Paxson V, Terrell J, "A Brief History of Scanning," *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*. ACM, 2007: 77-82.

[8] Durumeric Z, Wustrow E, Halderman J A, "ZMap: Fast Internet-Wide Scanning and Its Security Applications," *USENIX Security Symposium*. 2013, 8: 47-53.

[9] Xuan Feng, Qiang Li, Haining Wang, Limin Sun, "Acquisitional Rule-based Engine for Discovering Internet-of-Thing Devices", *27th {USENIX} Security Symposium ({USENIX}* Security 18) 2018, Baltimore, MD.

[10] Barnett R J, Irwin B, "Towards A Taxonomy of Network Scanning Techniques, " *Proceedings of the 2008 annual research conference of the South African Institute of Computer Scientists and Information Technologists on IT research in developing countries: riding the wave of technology. ACM*, 2008: 1-7.

[11] Shamsi Z, Nandwani A, Leonard D, et al, "Hershel: Single-Packet OS Fingerprinting," *ACM SIGMETRICS Performance Evaluation Review.* ACM, 2014, 42(1): 195-206.

[12] Caballero J, Venkataraman S, Poosankam P, et al, " FiG: Automatic Fingerprint Generation," *Department of Electrical and Computing Engineering*, 2007: 27.

[13] Kohno T, Broido A, Claffy K C, "Remote Physical Device Fingerprinting," *IEEE Transactions on Dependable and Secure Computing*, 2005, 2(2): 93-108.

[14] Yarochkin F V, Arkin O, Kydyraliev M, et al, "Xprobe2++: Low Volume Remote Network Information Gathering Tool," *Dependable Systems & Networks, 2009. DSN'09. IEEE/IFIP International Conference on*. IEEE, 2009: 205-210.

[15] M. Zalewski. (2008), "P0f Is the Passive Traffic Fingerprinting Tool," Available: http://lcamtuf.coredump.cx/p0f3/

[16] Auffret P. SinFP, "Unification of Active and Passive Operating System Fingerprinting," *Journal in computer virology*, 2010, 6(3): 197-205.

[17] Shamsi Z, Cline D B H, Loguinov D, "Faulds: A Non-Parametric Iterative Classifier for Internet-Wide OS Fingerprinting," *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017: 971-982.

[18] Richardson D W, Gribble S D, Kohno T, "the Limits of Automatic OS Fingerprint Generation," *Proceedings of the 3rd ACM workshop on Artificial intelligence and security*. ACM, 2010: 24-34.

[19] Whatweb Identifies Websites. Available: https://github.com/urbanadventurer/whatweb/wiki

[20] Wappalyzer Identify Technology on websites.Available: https://www.wappalyzer.com/

[21] Fachkha C, Bou-Harb E, Keliris A, et al, "Internet-Scale Probing of CPS: *Inference, Characterization and Orchestration Analysis," In Proceedings of the 2017 Network and Distributed System Security Symposium (NDSS 2017).*

[22] Feng X, Li Q, Wang H, et al. Characterizing Industrial Control System Devices on the Internet. *2016 IEEE 24th International Conference on Network Protocols (ICNP). IEEE*, 2016: 1-10.

[23] Cui A, Stolfo S J, "A Quantitative Analysis of the Insecurity of Embedded Network Devices: Results of a Wide-area Scan," *Proceedings of the 26th Annual Computer Security Applications Conference (ACSAC)*. ACM, 2010: 97-106.

[24] A. R. Khakpour, J. W. Hulst, Zihui Ge, A. X. Liu, Dan Pei and Jia Wang, "Firewall Fingerprinting," *IEEE INFOCOM 2012-The 31th Annual IEEE International Conference on Computer Communications.* Orlando, 2012, pp. 1728-1736.

[25] Radhakrishnan S V, Uluagac A S, Beyah R, "GTID: A Technique for Physical Device and Device Type Fingerprinting," *IEEE Transactions on Dependable and Secure Computing*, 2015, 12(5): 519-532.

[26] Li Q, Feng X, Wang H, et al, "Automatically Discovering Surveillance Devices in the Cyberspace," *Proceedings of the 8th ACM on Multimedia Systems Conference.* ACM, 2017: 331-342.

[27] Q. Li, X. Feng, H. Wang, Z. Li, and L. Sun, "Towards

fine-grained fingerprinting of firmware in online embedded devices," *in IEEE International Conference on Computer Communications (INFOCOM)*. IEEE, 2018

[28] Padmanabhan R, Dhamdhere A, Aben E, et al, "Reasons Dynamic Addresses Change," *Proceedings of the 2016 ACM on Internet Measurement Conference*. ACM, 2016: 183-198.

[29] Zander S, Murdoch S J, "An Improved Clock-skew Measurement Technique for Revealing Hidden Services," *USENIX Security Symposium*. 2008: 211-226.

[30] Feng X, Li Q, Han Q, et al, "Active Profiling of Physical Devices at Internet Scale," *Computer Communication and Networks (ICCCN), 2016 25th International Conference on*. IEEE, 2016: 1-9.

[31] Formby D, Srinivasan P, Leonard A, et al, "Who's in Control of Your Control System? Device Fingerprinting for Cyber-Physical Systems," *In Proceedings of the 2016 Network and Distributed System Security Symposium (NDSS 2016)*.

[32] Urbina D I, Giraldo J A, Cardenas A A, et al, "Limiting the Impact of Stealthy Attacks on Industrial Control Systems," *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016: 1092-1105.

[33] Beverly R, "Yarrp'ing the Internet: Randomized High-Speed Active Topology Discovery," *Proceedings of the 2016 ACM on Internet Measurement Conference*. ACM, 2016: 413-420.

[34] Bartlett G, Heidemann J, Papadopoulos C, "Understanding Passive and Active Service Discovery," *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*. ACM, 2007: 57-70.

[35] Nessus, Online Vulnerability Scanner, [Online]. Available: http://www. nessus.org

[36] Smith F D, Campos F H, Jeffay K, et al, "What TCP/IP Protocol Headers Can Tell Us About the Web," *ACM SIGMETRICS Performance Evaluation Review*. ACM, 2001, 29(1): 245-256.

[37] Bailey M, Cooke E, Jahanian F, et al, "The Internet Motion Sensor-A Distributed Blackhole Monitoring System," *In Proceedings of the 2005 Network and Distributed System Security Symposium (NDSS 2005)*.

[38] CAIDA and Telescope, the UCSD Network Telescope and Center for the Applied Internet Data Analysis. [Online]. Available: http://www.caida.org/ projects/network_telescope/

[39] Moore D, Shannon C, Voelker G M, et al, "Network Telescopes: Technical Report," Department of Computer Science and Engineering, University of California, San Diego, 2004.

[40] Glatz E, Dimitropoulos X, "Classifying Internet One-way Traffic," *Proceedings of the 2012 ACM conference on Internet measurement conference*. ACM, 2012: 37-50.

[41] Gates C, "Coordinated Scan Detection," *In Proceedings of the 2009 Network and Distributed System Security Symposium (NDSS 2009)*.

[42] DWhyte D, Kranakis E, Van Oorschot P C, "DNS-based Detection of Scanning Worms in an Enterprise Network," *In Proceedings of the 2005 Network and Distributed System Security Symposium (NDSS 2005)*.

[43] Heidemann J, Pradkin Y, Govindan R, et al, "Census and Survey of the Visible Internet," *Proceedings of the 8th ACM SIGCOMM conference on Internet measurement*. ACM, 2008: 169-182.

[44] Leonard D, Yao Z, Wang X, et al, "Stochastic Analysis of Horizontal IP Scanning," *IEEE INFOCOM 2012-The 31th Annual IEEE International Conference on Computer Communications*. 2077-2085.

[45] Durumeric Z, Bailey M, Halderman J A, "An Internet-Wide View of Internet-Wide Scanning," *USENIX Security Symposium*. 2014: 65-78.

[46] Leonard D, Loguinov D, "Demystifying Service Discovery:Implementing an Internet-wide Scanner," *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*. ACM, 2010: 109-122.

[47] François J, Abdelnur H, Festor O, "Semi-supervised Fingerprinting of Protocol Messages," *Computational Intelligence in Security for Information Systems 2010*. Springer, Berlin, Heidelberg, 2010: 107-115.

[48] François J, Abdelnur H, State R, et al, "Machine Learning Techniques for Passive Network Inventory," *IEEE Transactions on Network and Service Management,* 2010, 7(4): 244-257.

[49] Xu Z, Nappa A, Baykov R, et al, "Autoprobe: Towards Automatic Active Malicious Server Probing Using Dynamic Binary Analysis," *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2014: 179-190.

[50] Caballero J, Yin H, Liang Z, et al, "Polyglot: Automatic Extraction of Protocol Message Format using Dynamic Binary Analysis," *Proceedings of the 14th ACM conference on Computer and communications security*. ACM, 2007: 317-329.

[51] Wondracek G, Comparetti P M, Kruegel C, et al, "Automatic Network Protocol Analysis," *In Proceedings of the 2008 Network and Distributed System Security Symposium (NDSS 2008)*. p1-14.

[52] Brumley D, Caballero J, Liang Z, et al, "Towards Automatic Discovery of Deviations in Binary Implementations with Applications to Error Detection and Fingerprint Generation," *USENIX Security Symposium*. 2007: 15.

[53] Comparetti P M, Wondracek G, Kruegel C, et al, "Prospex:

Protocol Specification Extraction," *Security and Privacy,* 2009 30th IEEE Symposium on. IEEE, 2009: 110-125.

[54] Leonard D, Loguinov D, "Demystifying Service Discovery: Implementing an Internet-wide Scanner," *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement.* ACM, 2010: 109-122.

[55] Botnet C, "Internet Census 2012: Port Scanning/0 Using insecure Embedded Devices," Abstract [Online]. Available: http://internetcensus2012. bitbucket. org/paper. html, 2013.

[56] Krishnamurthy B, Wang J, "On Network-Aware Clustering of Web Clients," *ACM SIGCOMM Computer Communication Review,* 2000, 30(4): 97-110.

[57] Cai X, Heidemann J, "Understanding Block-level Address Usage in the Visible Internet," *ACM SIGCOMM Computer Communication Review.* ACM, 2010, 40(4): 99-110.

[58] Hong C Y, Yu F, Xie Y, "Populated IP Addresses: Classification and Applications," *Proceedings of the 2012 ACM conference on Computer and communications security.* ACM, 2012: 329-340.

[59] Xu K, Wang F, Gu L, "Network-aware Behavior Clustering of Internet End Hosts," IEEE INFOCOM 2011, *the 30th Annual IEEE International Conference on Computer Communications,* 2078-2086.

[60] Quan L, Heidemann J, Pradkin Y, "Trinocular: Understanding Internet Reliability Through Adaptive Probing," *ACM SIGCOMM Computer Communication Review.* ACM, 2013, 43(4): 255-266.

[61] Springall D, Durumeric Z, Halderman J A, "FTP: The Forgotten Cloud. Dependable Systems and Networks (DSN)," *2016 46th Annual IEEE/IFIP International Conference on.* IEEE, 2016: 503-513.

[62] Staniford S, Paxson V, Weaver N, "How to Own the Internet in Your Spare Time," *USENIX Security Symposium*. 2002, 2: 14-15.

[63] Krupp J, Backes M, Rossow C, "Identifying the Scan and Attack Infrastructures Behind Amplification DDoS Attacks," *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016: 1426-1437.

[64] Dainotti A, King A, Papale F, et al, "Analysis of a/0 Stealth Scan from A Botnet," *Proceedings of the 2012 ACM conference on Internet measurement conference.* ACM, 2012: 1-14.

[65] Cui A, Costello M, Stolfo S J, "When Firmware Modifications Attack: A Case Study of Embedded Exploitation," *In Proceedings of the 2013 Network and Distributed System Security Symposium (NDSS 2013)*.

[66] Durumeric Z, Kasten J, Adrian D, et al, "The Matter of Heartbleed," *Proceedings of the 2014 Conference on Internet Measurement Conference.* ACM, 2014: 475-488.

[67] Aryan S, Aryan H, Halderman J A, "Internet Censorship in Iran: A First Look," *the 3rd Workshop on Free and Open Communications on the Internet*. Washington, USA, 2013.

[68] Nappa A, Xu Z, Rafique M Z, et al, "Cyberprobe: Towards Internet-scale Active Detection of Malicious Servers," *In Proceedings of the 2014 Network and Distributed System Security Symposium (NDSS 2014)*. 2014: 1-15.

[69] Zhang J, Hu X, Jang J, et al, "Hunting For Invisibility: Characterizing and Detecting Malicious Web Infrastructures through Server Visibility Analysis," *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*. 2016: 1-9.

[70] Liao X, Alrwais S, Yuan K, et al, "Lurking Malice in the Cloud: Understanding and Detecting Cloud Repository as a Malicious Service," *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security.* ACM, 2016: 1541-1552.

李强 于 2015 年在中国科学院计算机科学专业获得博士学位。现任北京交通大学计算机与信息技术学院助理教授。研究领域为物联网安全。研究兴趣包括: 移动计算、物联网。

Email: liqiang@bjtu.edu.cn

贾煜璇 于2016年在河北工业大学计算机科学与技术专业获得学士学位。现在北京交通大学学校网络空间安全专业攻读硕士学位。研究领域为物联网安全。研究兴趣包括: 物联网、信息安全。

Email: jyxuan@bjtu.edu.cn

宋金珂 于2017年在北京交通大学计算机与信息技术学院计算机与科学技术专业获得工学硕士学位, 现在北京交通大学网络空间安全专业攻读博士学位。研究领域为网络空间安全、物联网安全。研究兴趣包括: 网络空间安全、物联网安全、信息安全。Email: 17112099@bjtu.edu.cn

朱红松 现任中国科学院信息工程研究所物联网安全实验室研究员, 计算机学会高级会员, 传感器网络专业委员会委员。研究领域为物联网安全、网络攻防、安全大数据分析与测评。
Email: zhuhongsong@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。研究领域为物联网安全、区块链安全。研究兴趣包括: 物联网安全、区块链安全。Email: lihong@iie.ac.cn

孙立民 现任中国科学院信息工程研究所教授, 计算机科学杂志和计算机应用杂志的编辑, EURASIP 无线通信和网络杂志以及网络期刊特刊的客座编辑。研究兴趣包括: 移动车辆网络、延迟容忍网络、无线传感器网络、移动 IP 以及下一代互联网技术。