智能家居攻击与防御方法综述

严 寒[1,2], 彭国军[1,2], 罗 元[1,2], 刘思德[1,2]

[1] 武汉大学 空天信息安全与可信计算教育部重点实验室 武汉 中国 430072
[2] 武汉大学 国家网络安全学院 武汉 中国 430072

摘要 智能家居是物联网的一大发展方向, 但其在安全方面表现得不如人意, 近年来频频爆发网络安全事件。智能家居相较于传统的嵌入式设备, 引入了移动应用程序和云平台服务, 使得其暴露出了更多的攻击面。本文围绕智能家居终端设备、云平台、移动应用程序及通信等四个方面, 综述针对智能家居的攻击方法和防御措施, 并针对性的梳理了目前学术界及工业界关注的研究热点与难点。最后, 本文针对现有智能家居设备自动化漏洞挖掘技术与防御监控能力的不足进行了讨论, 并提出了基于 Docker 集群部署的端侧自动化威胁模型系统设计思路。

关键词 物联网; 智能家居; 攻击; 防御; 漏洞
中图法分类号 TP309.1 DOI 号 10.19363/J.cnki.cn10-1380/tn.2021.07.01

Survey on Smart Home Attack and Defense Methods

YAN Han[1,2], PENG Guojun[1,2], LUO Yuan[1,2], LIU Side[1,2]

[1] Key Laboratory of Aerospace Information Security and Trust Computing, Ministry of Education, Wuhan University, Wuhan 430072, China
[2] School of Cyber Science and Engineering, Wuhan University, Wuhan 430072, China

Abstract The smart home is a major development direction of the Internet of Things, but its performance is not satisfying in terms of security. In recent years, network security events erupted repeatedly. Compared with traditional embedded devices, smart home introduces mobile applications and cloud platforms, thus exposing more attack surfaces. This paper focuses on four aspects: smart devices, cloud platforms, mobile applications, and communications. We summarize the attack and defense methods for smart home and summarize the current research hotspots and difficulties between academia and industries. Finally, this paper discusses the limitations of existing automation vulnerability mining and defensive monitoring capabilities of the smart home. Based on these efforts, we propose the design concept of end-side automated threat model system based on Docker cluster deployment.

Key words the Internet of Things; smart home; attack; defense; vulnerability

1 引言

物联网被人们视作继计算机、互联网之后信息技术产业的第三次革命, 在泛在化物联网构建的场景中, 人与物之间跨越了时间和空间的约束, 被紧紧联接在一起。根据麦卡锡公司[1]的分析估计, 到 2025 年, 物联网(the Internet of Things, IoT)每年潜在的经济总影响为 3.9 万亿至 11.1 万亿美元, 其中, 智能家居作为物联网的一大发展方向, 经济规模将达到 2 千亿至 3 千亿美元。尚在蓬勃发展中的智能家居市场被传统的家居制造企业和新兴的互联网公司视作金矿, 各大厂商纷纷研发推出自己的智能家居系统, 并向消费者提供配套的智能设备、云平台及移动应用程序, 国外厂商如三星的 SmartThings[2]、亚马逊的 AWS[3]、苹果的 HomeKit[4]、谷歌的 Weave/Brillo[5]和微软的 Azure[6], 国内主要有阿里云 IoT[7]、华为 HiLink[8]和小米 MiJia[9]。

在智能家居产业飞速发展的过程中, 其安全状态却不容乐观。厂商急于开发出新功能和新产品吸引消费者从而扩大智能家居市场份额, 但受到产品上市时间以及研发成本的限制, 安全人员无法在开发周期中投入足够多和广泛的安全测试工作。这些产品安全的疏忽加上智能家居潜在的经济效应, 吸引了许多黑客组织对智能家居系统进行漏洞挖掘和利用。

针对智能家居系统最常见的有两种攻击场景,

通讯作者: 彭国军, 教授, Email: guojpeng@whu.edu.cn。
本课题得到 NSFC-通用技术基础研究联合基金(No. U1636107); 国家自然科学基金(No. 61972297)资助。
收稿日期: 2020-10-20; 修改日期: 2020-12-24; 定稿日期: 2021-06-24

一种是黑客利用设备漏洞对设备进行远程控制或者获取智能家居系统中的隐私数据, 比如 Nest 恒温器在主人不知情的情况下打开了摄像头[10]; 黑客通过飞利浦婴儿摄像头的漏洞可以监控婴儿[11]。除对用户的隐私造成侵犯之外, 由于智能家居设备具有对物理世界的操作功能, 非法的攻击行为甚至可能对消费者的财产及生命安全带来威胁, 比如攻击者可以对智能门锁进行控制操作, 从而非法闯入用户的家, 窃取物理财产。

另一种是利用设备安全缺陷控制大规模的设备, 形成僵尸网络从而发动分布式拒绝服务攻击(Distributed Denial of Service, DDoS), 比如2016年席卷全美国的“Mirai 僵尸网络[12]”以及后来的效仿者, 就利用了供应商留下的后门[14-15]、不安全的应用服务以及许多设备暴露在公网之上的弱点, 在设备之间快速传播恶意代码, 利用大量的智能家居设备形成的僵尸网络, 造成网络瘫痪、设备拒绝服务等严重后果, 对社会的经济活动及人们的日常生产生活造成极大的损失。

供应商及安全研究人员已经意识到问题的严重性, 采取一系列的措施来保障智能家居的安全, 但智能家居安全所面临的最大挑战在“大”“多”“杂”。智能家居终端节点组成的网络巨大、设备数量繁多, 因此智能设备每天都会采集大量异质数据交给云平台进行处理; 尽管大多数设备都是基于嵌入式 Linux 操作系统的, 但设备的特定功能造成设备运行的服务、网络协议、硬件配置都有所差异, 这些差异使得安全研究人员很难采取统一的安全分析和防护手段来保护智能家居产品。

相较于传统 PC 及嵌入式设备, 智能家居拥有更复杂的应用场景。总体来说, 智能家居系统可以分为云平台、通信管道、终端设备以及移动应用程序四个层面。云平台作为整个生态系统的中枢大脑, 其主要安全问题包括平台逻辑缺陷、权限粒度过粗、语音识别漏洞、用户隐私保护和设备恢复与诊断; 终端设备是整个系统中最不可信的一端, 而固件安全又是设备安全的基石, 由此产生了大量针对固件提取以及固件安全启动与更新的攻击和防御手段, 此外, 设备用于采集数据的传感器也是黑客关注的焦点; 在移动应用程序方面, 权限、数据存储及编程质量存在着大量安全问题; 作为连接三个端点的管道, 通信协议错综复杂, 主要有明文传输及中间人攻击两大风险。

需要指出的是, 目前国内外的学术界及工业界对智能家居的攻击及防御方法有很多的研究工作, 但这些工作较为分散, 本文将系统地梳理研究脉络和重点, 对研究热点和难点进行整体论述和分析, 为学术界和相关从业者提供一定的参考。

本文主要有如下两部分的工作和贡献:

(1) 从智能家居设备、配套的移动应用程序、云平台和相关的通信管道等四个方面, 梳理目前针对智能家居系统的攻击技术和防御措施, 并针对性地分析安全现状及梳理安全发展历程。

(2) 对与物联网安全和隐私相关的学术研究进行梳理分析, 包括近年来学术界的安全研究人员关注的热点话题以及技术难点, 主要介绍针对终端设备的自动化漏洞挖掘技术发展情况, 以及主流物联网平台面对动态设备威胁时所采用的防御手段, 并对现有工作的不足之处进行讨论, 提出了在安全最佳实践快速迭代情况下端侧自动化威胁模型系统的设计思路。

本文的组织结构如下: 第 2 章结合智能家居应用模型论述智能家居安全现状, 包括物联网中常见的漏洞类型及恶意程序; 第 3 章分别从应用模型的四个方面进行攻击与防御技术的综述; 第 4 章对当前物联网安全领域的热点及技术难点进行讨论, 并提出端侧自动化威胁模型的设计思路; 第 5 章总结全文并展望。

2 智能家居安全现状

智能家居在高速发展及大规模应用的过程中, 频频爆发出影响巨大的安全事件, 这也引发了民众对个人隐私数据及资产的担忧。

本节将提出智能家居应用模型, 从智能家居与传统互联网和计算机不同的特性出发, 对典型的物联网漏洞类型与恶意代码进行总结和分析, 阐明智能家居设备的安全现状。

2.1 智能家居应用模型

信息物理系统(Cyber Physical Systems, CPS)和 IoT 在其解决方案和体系结构中有很多相似性, 但具体实现和侧重点有所差异。CPS 是一个综合了计算、网络和物理环境的多维复杂系统, 用户通过人机交互接口来监督和控制物理环境, 操作结果由物理环境反馈给计算机, 正如互联网改变了人类彼此交互的方式一样, 网络物理系统也将改变我们与周围物理世界的交互方式。IoT 与 CPS 的区别在于 CPS 不一定连接到互联网(Internet)中, IoT 强调设备之间的连接性, 而 CPS 强调嵌入式部分。智能家居是 IoT 的子集, 是 IoT 在家庭领域的具体应用场景, 家庭用户和云平台的出现, 使得智能家居相较于传统 IoT,

时刻面临着新的复杂攻击, 例如隐私泄漏和分布式僵尸网络等。因此, 智能家居安全需要研究者从不同于以往CPS和IoT安全的角度出发, 来发现问题和设计防护方案。本文主要总结和分析智能家居领域的攻击方法和防御措施。

为了以统一的方式管理数量不断增长的各种智能家居设备, 许多公司提出了他们的智能家居平台, 各个厂商设计的智能家居系统各有差异, 但通过对国内外智能设备云平台的研究分析, 可以抽象出统一的拓扑结构, 如图 1 所示。

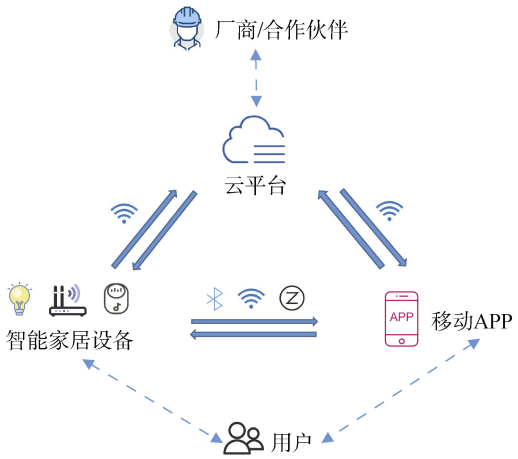


图 1 智能家居系统部署拓扑图

Figure 1 Topology diagram of smart home system deployment

拓扑图里面包含三个交互的实体: 云平台、智能家居终端设备、移动应用程序, 三个端点之间通过Wi-Fi、蓝牙等通信管道传输数据, 其中用户对终端设备和 APP 有直接的控制权, 云平台还会通过 API 接口向厂商及第三方伙伴提供服务。各实体功能主要如下:

云平台: 云平台是智能家居系统的大脑。它主要承担四个功能, 分别是: 设备绑定、设备命令控制、家庭自动化以及开发者管理。首先, 设备绑定需要在设备首次部署时建立所有者账户与所有设备之间一一映射的关系, 以保证只有合法的授权用户可以对设备进行命令控制; 其次, 授权用户向设备发送的控制命令需要通过云平台进行处理和转发; 另外, 大多数智能家居系统提供家庭自动化服务, 用户可以定义一系列事件规则以完成自动化, 比如当接近下班时间时, 房间的空调自动打开, 用户回到家后无需再等待温度降低的过程; 最后, 设备厂商可以通过云平台对设备做统一管理, 比如下发固件更新、异常日志处理、大数据态势感知等。

智能家居终端设备: 常见的智能家居设备包括: 摄像头、路由器、音箱、灯泡、门锁等。以智能音箱为例, 其硬件架构如图 2 所示, 设备通常配置了各种传感器从物理世界采集数据, 在本地简单格式化之后将数据发送给云平台, 并将云平台反馈的信息显示给用户。设备有两种典型的通信管道可以连接到云平台: (a)配有无线网卡的设备可以通过Wi-Fi与云平台直接通信; (b)没有配置 Wi-Fi 接口的节能型设备, 会通过 BLE 或 ZigBee 等协议连接到移动 APP 或智能网关, 然后网关和移动 APP 转发连接到云平台。

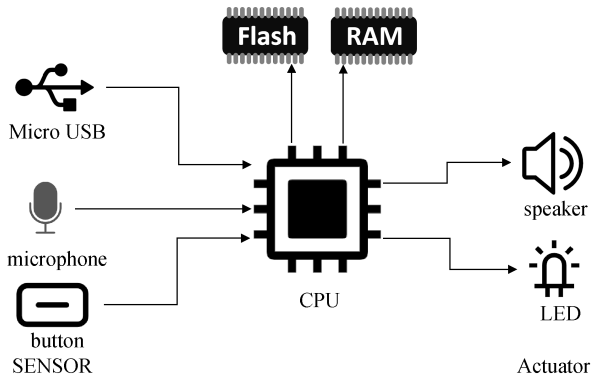


图 2 智能音箱硬件架构

Figure 2 The hardware architecture of smart speaker

移动应用程序: 移动 APP 为家庭用户提供了友好的交互界面, 用户可以通过它完成绑定设备、远程管理设备、定义自动化规则等功能。

2.2 智能设备典型漏洞

由于部分智能家居的开发者缺乏安全意识, 设备的保护措施非常脆弱而且漏洞频出。如图 3 所示, 本文以通信层关键设备—路由器为例, 对通用漏洞披露(Common Vulnerabilities & Exposures, CVE)中的历年漏洞记录数量进行了统计。自 2013 年智能家居

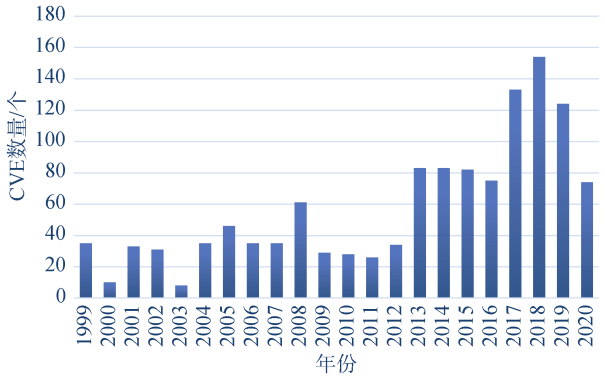


图 3 CVE 披露的历年路由器漏洞记录数量

Figure 3 Number of router vulnerability records disclosed by CVE in recent years

的概念进入路由器行业之后, 路由器的漏洞记录数量较往年有明显增长, 在最近三年, 每年的漏洞数量都突破了 120 个。智能路由器的快速发展并没有提高设备的安全性, 相反导致了漏洞数量的猛增, 给使用智能路由器的家庭用户带来了巨大的安全风险。

国家信息安全漏洞共享平台(CNVD)2019 上半年公开收录智能设备安全漏洞 1223 个[16], 与 2018 年同期基本持平。这些安全漏洞涉及的漏洞类型主要包括设备信息泄露、权限绕过、远程代码执行、弱口令等; 涉及的设备类型主要包括家用路由器、网络摄像头等。智能家居的漏洞成因复杂多样, 通常将智能家居中最常见的漏洞细分为十个类别[18], 下文将结合智能家居应用模型、系统特性和利益相关方责任, 通过典型的漏洞实例来对这十种漏洞进行总结分析。

(1) 弱密码、可猜测密码或硬编码密码: 由于供应商及消费者的安全意识不足, 一些设备的远程管理服务的初始凭据安全强度很低, 而用户很少修改默认密码, 造成许多僵尸网络的感染途径都是使用暴力破解密码或默认出厂密码, 通过 SSH/Telnet 登录入侵物联网设备, 取得设备的控制权, 比如“Mirai”僵尸网络。这种凭证配置不当的情况是普遍存在的, Deepak Kumar 等人[19]使用由一些弱凭证或默认配置密码组成的小型字典尝试登录 FTP 和 Telnet 服务, 从而识别出使用弱/默认凭证进行身份验证的物联网设备, 7.1%的 IoT 设备和 14.6%的家用路由器开启了 FTP 和 Telnet 服务, 其中有 17.4%的用户设置了较弱的 FTP 凭证, 有 2.1%的用户使用了较弱的 Telnet 凭证。最为流行的弱凭证如表 1 所示, admin/admin 是最常见的, 分别占了 88%和 36%。

表 1 最流行的 FTP 和 Telnet 弱凭证

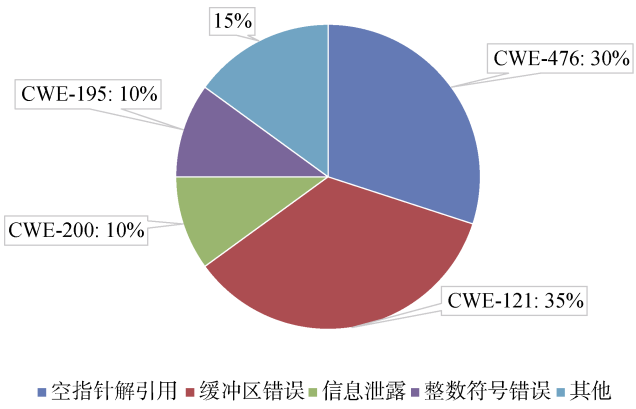
Table 1 The most popular FTP and Telnet weak credentials

FTP 凭证	比例(%)	Telnet 凭证	比例(%)
admin/admin	88.3	admin/admin	35.6
admin/	5.9	root/xc3511	16.0
Administrator/	1.4	vodafone/vodafone	10.4
sysadm/sysadm	0.9	guest/guest	7.8
root/	0.7	admin/1234	7.5
root/root	0.4	root/hslwificam	3.9
user/	0.4	root/vizxv	3.7
meo/meo	0.3	root/oelinux123	2.2
admin/password	0.3	admin/4321	1.8
admin/ttnet	0.3	-/-	1.6
other	1.0	other	9.5

(2) 使用不安全或已遭弃用的组件: 智能家居设备通常会使用定制的开源操作系统平台以及第三方的软件或硬件组件, 由于开源代码更新较快而且源代码公开, 而供应商对已经出现的问题的过期组件更新不及时, 因此攻击者可以采用低版本的已知漏洞对设备进行攻击, 典型的过期组件有 busybox、openssl、ssh、Mini_httpd 等。2018 年 Mini_httpd 组件被爆出任意文件读取漏洞(CVE-2018-18778[26]), 据估测该漏洞影响全球两百多万台设备; 2020 年, 轻量级 TCP/IP 软件库被披露出 19 个 0 day 漏洞(Ripple 20[27]), 可能导致不同行业的数十亿 IoT 设备面临着远程攻击的风险。

(3) 不安全的网络服务: 设备运行的网络服务多样, 比如 UPnP[20]、Telnet、Samba[21], 但供应商在实现这些服务类型时很少完全遵循协议和安全编程规范, 将端口暴露在互联网上, 很容易遭到 DoS 或远程代码执行攻击, 对设备的安全性造成威胁, 如运行在华为 HG532 系列路由器[22]上的 UPnP 服务出现命令执行漏洞(CVE-2017-17215); 为 IoT 设备提供文件共享服务的 Samba 被爆出 Linux 版“永恒之蓝[23]”攻击(CVE-2017-7494)。

嵌入式设备中大多使用轻量级的 MiniUPnP 库来实现 UPnP 协议, 自 2013 年开始, 共披露出 20 条漏洞条目, 本文统计其漏洞类别分布如图 4 所示, 从分布情况可以看出, 缓冲区错误及空指针解引用问题是物联网二进制程序中比较典型的两大代码缺陷。



(注: 通用缺陷列表(Common Weakness Enumeration, CWE)是一个对软件脆弱性和易受攻击性的一个分类系统)

图 4 MiniUPnP 漏洞类别分布

Figure 4 MiniUPnP vulnerability category distribution

(4) 隐私保护不充分: 与其他网络设备相比, 智能家居设备配置了大量传感器采集用户的生物信

息、监测和记录周围环境信息和用户活动情况, 消费者和智能家居之间的亲密关系导致在设备或云平台中存储着用户的大量个人信息, 一旦被不安全的、不当的或未经授权的使用, 会对用户的隐私造成危害, 比如窃取私人照片、泄露用户位置、窃听用户输入, 典型攻击案例如表 2 所示。

表 2 隐私保护典型案例

Table 2 Typical case of privacy protection

设备名	CWE	安全影响
Gator2 smartwatch[28]	CWE-359	攻击者可以访问包含软件版本、IMEI、时间、定位方法(GPS 与 Wi-Fi)、位置坐标、电池电量等信息。
三星智能电视[29]	CWE-359	攻击者可通过名为“Weeping Angel”的工具获取记录的音频数据, 还可以伪装成电视屏幕关闭继续监听。
Yeelight 智能 AI 音箱[30]	CWE-200	攻击者可以窃听音频数据, 读取日志文件中的明文 Wi-Fi 凭据或访问其他敏感设备和用户信息。

(5) 不安全的生态接口: 智能家居系统使用 Web、云端或移动接口供各个组件之间交互, 攻击者通常会在智能设备的 Web 接口中寻找 XSS、CSRF 和 SQLi 等漏洞, 以及在云和移动 APP 接口中寻找“弱密码”“信息泄露”“缺乏双因子认证”和“无账户锁定机制”等问题。亚马逊的 Ring Video Doorbell Pro 使用了不安全的配网模式, 如图 5 所示[24], 附近的攻击者通过无线嗅探或监听可以截获网络凭证, 从而进入局域网发起进一步的攻击。

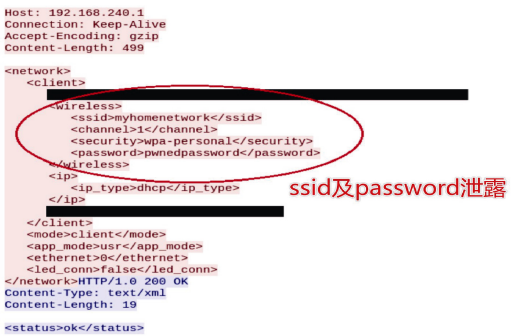


图 5 亚马逊门锁配网模式下信息泄露

Figure 5 Information leakage in Amazon door lock under distribution network mode

(6) 缺乏安全的更新机制: 固件安全是设备安全的基石。受到资源的限制, 许多基于密码学的安全功能难以直接应用到智能家居领域, 如基于数字签名的安全升级。2017 年 D-Link DIR8xx 系列路由器被发现升级逻辑不当[25], 设备没有对固件更新进行相应的完整性和合法性校验, 攻击者可以向设备安装任意官方版本或自定义的固件, 从而接管设备的控制权。我们对 CVE 记录的相关漏洞进行了统计, 结果显示针对固件更新的攻击在 2018 年达到了峰值, 而且到目前为止依然有厂商采取不安全的更新机制。

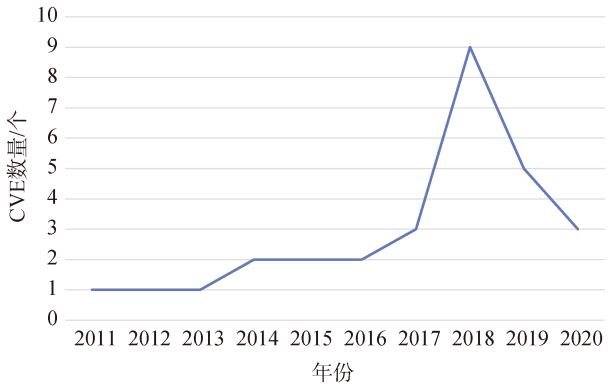
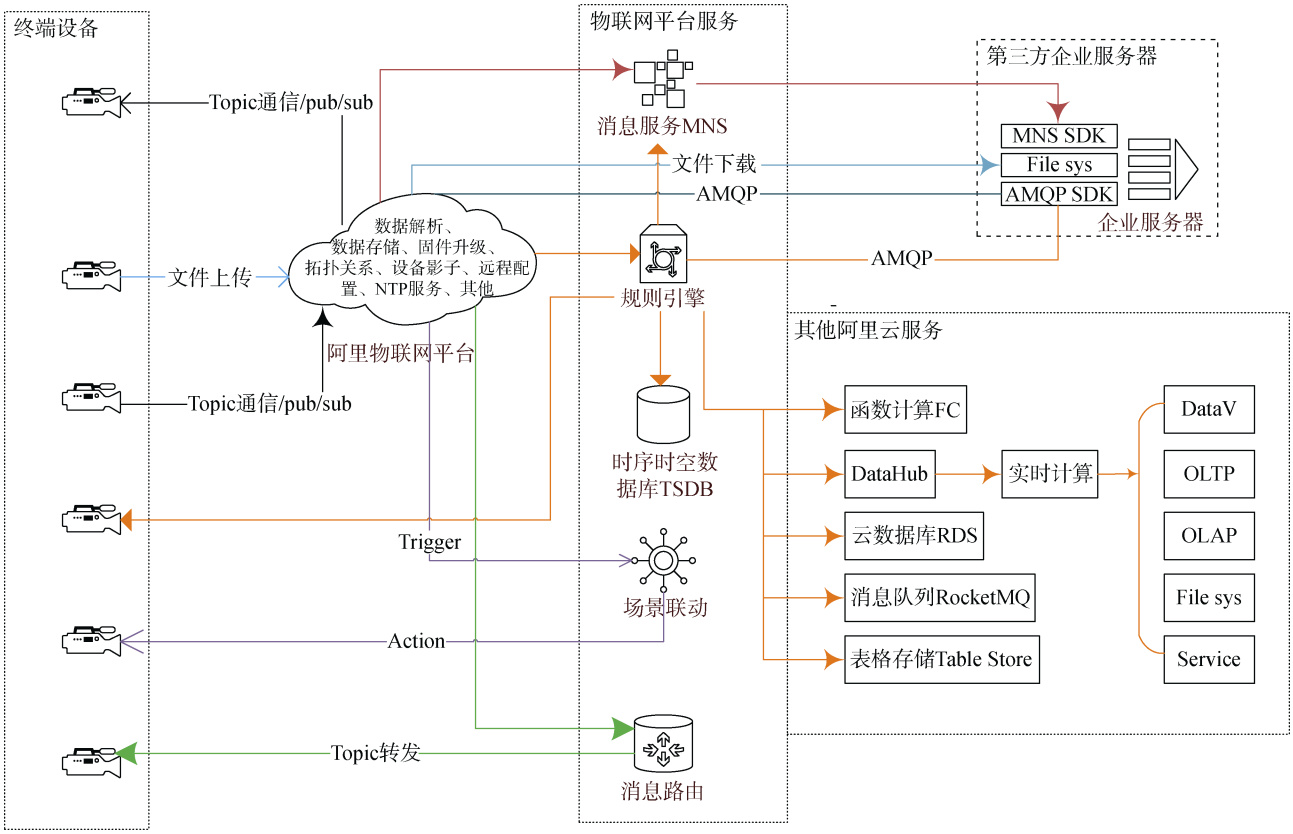


图 6 CVE 披露的固件更新漏洞历年记录数量

Figure 6 Number of firmware update vulnerabilities disclosed by CVE in recent years

(7) 不安全的数据传输和存储: 智能家居系统各个组件存储着用户的敏感数据, 并通过各种通信管道传输各自的数据内容。Gartner 报告显示[31], 物联网市场依然保持着 3A 格局(亚马逊、微软、阿里云), 其中阿里云借助中国广大的市场以及在云服务上的技术优势, 向工业、医疗、交通、城市、航空等领域提供了百亿级 IoT 设备的连接能力和上万个行业解决方案, 因此本文对阿里云物联网平台上设备数据在各个组件间的流转途径进行了总结, 如图 7 所示。由于计算资源和存储资源的限制, 许多智能家居设备都没有检查服务端的证书甚至不对通信过程加密, 复杂的加密和认证算法会占用过多的计算资源, 降低设备的性能, 影响设备的正常运行, 对于实时设备是不可忍受的。如果敏感数据内容明文传输或没有以正确的方式进行加密保护, 攻击者常常会对其进行中间人攻击, 从而获取传输内容, 比如 Philips 婴儿监控摄像头没有提供防止攻击者窃听的加密视频流[11]。



(注: 物联网平台中, 服务端和设备端通过 Topic 来实现消息通信, 设备具有发布(pub)和订阅(sub)两种操作权限)

图 7　阿里云 IoT 平台数据流转图

Figure 7　Data flow diagram of Aliyun IoT platform

(8) 缺乏设备管理: 智能家居设备地理分布位置广泛, 供应商对已经投入市场的智能家居设备缺乏安全支持, 比如资产管理、更新管理、安全解除、系统监控和响应能力, 一旦设备出现故障或漏洞, 管理人员无法对错误进行排查、分析和处理。

(9) 不安全的默认配置: 供应商虽然提供了安全的机制, 但默认配置为不安全的策略。用户的安全意识通常是薄弱的, 导致安全机制无法充分地发挥作用, 比如谷歌提供的双因素验证可以消除密码泄露带来的撞库风险, 但用户对这项安全机制一无所知, 因此黑客利用其他网站上的密码泄露, 入侵并控制了谷歌的智能家居设备 Smart Nest Thermostat[10]。

(10) 缺乏物理加固措施: 智能设备的硬件设计远没有计算机和智能手机复杂, 这大大降低攻击者通过硬件分析发起攻击的难度。电路板上暴露其 MCU、外部存储器等信息, 攻击者拆开设备之后通过查询 datasheet 就可以掌握设备的硬件情况。之后, 通过 JTAG 调试接口或 UART 等串口, 攻击者还可以对固件内容进行相应的读写操作。此外, 还可以通过硬件攻击绕过软件保护, 比如 Nest 恒温器硬件基础设施缺乏合适的保护[32], 攻击者可以通过改变设备引导过程绕过固件更新校验。

智能家居系统有着区别于其他联网设备的独特属性。智能家居设备类别繁多、功能各异、交互复杂, 催生出了许多攻击面, 再加上智能设备硬件设计简单, 且存储和计算资源无法支撑传统安全的一些防御措施, 导致智能家居相对于传统平台上的产品更容易被攻击者入侵, 由于智能设备通常与家庭物理环境紧密联系, 通过传感器采集信息并利用执行器(如加热器、加湿器)影响物理环境, 智能家居的安全性问题往往会造成严重后果。这些特性为智能家居系统带来了潜在的威胁和漏洞, 表 3 将智能家居中常见的漏洞类型与其相关的特性联系起来, 结合 2.1 节对智能家居应用模型的讨论, 确定了其影响的系统层面, 并归属了利益相关方的责任, 其中考虑到移动应用安全规范与生态已经较为独立与成熟, 本文将“端”安全分离为终端设备安全以及移动应用程序安全。

2.3　恶意代码攻击威胁

在互联网时代, 计算机是恶意病毒、蠕虫入侵的主要对象[33]。随着物联网的发展, 智能设备走进千家万户, 黑客逐渐把入侵的对象从主机转向安全性更

表 3 智能家居漏洞类型及其涉及的特性、系统层面和利益相关方责任

Table 3 The features, system level and stakeholder responsibilities of each smart home vulnerabilities

漏洞类型	特性	相关层面				利益相关方	
		云	管	设备端	APP 端	消费者	供应商
弱身份验证	意识不足	√		√	√	√	√
不安全的组件	更新滞后		√	√			√
不安全的网络服务	协议多样			√			√
隐私保护不充分	关系亲密	√				√	√
不安全的生态接口	交互复杂	√		√	√		√
缺乏安全的更新机制	资源有限			√			√
不安全的数据传输和存储	资源有限	√	√				√
缺乏设备管理	分布广泛	√					√
不安全的默认配置	意识不足	√	√	√	√	√	√
缺乏物理加固措施	硬件简单			√			√

低的智能设备。这些恶意程序的感染途径一般是通过软件漏洞利用以及暴力破解凭证来入侵和远程控制设备。设备被黑客入侵控制后, 通常会成为僵尸网络的一员, 被用来执行 DDoS 攻击或其他恶意行为。2019 年, CNCERT 捕获智能设备恶意程序样本约 324.1 万个[34], 其中 Mirai 家族和 Gafgyt 家族的恶意样本就占据了 86.1%, 被控制的智能设备平均每天会对 1528 个目标主机发起 DDoS 攻击。根据卡巴斯基安全报告[35], 攻击者在破解路由器 telnet 密码之后使用最多的十种恶意软件如表 4 所示。

表 4 黑客入侵路由器最常使用的十种恶意软件

Table 4 The ten most commonly used malware to attack routers

排名	恶意代码名	占比(%)
1	Backdoor.Linux.Mirai.c	15.97
2	Trojan-Downloader.Linux.Hajime.a	5.89
3	Trojan-Downloader.Linux.NyaDrop.b	3.34
4	Backdoor.Linux.Mirai.b	2.72
5	Backdoor.Linux.Mirai.ba	1.94
6	Trojan-Downloader.Shell.Agent.p	0.38
7	Trojan-Downloader.Shell.Agent.as	0.27
8	Backdoor.Linux.Mirai.n	0.27
9	Backdoor.Linux.Gafgyt.ba	0.24
10	Backdoor.Linux.Gafgyt.af	0.20

Mirai 家族的恶意代码构建的僵尸网络至今已参与了多次的大型分布式拒绝服务攻击(DDoS), 最早也是最受关注的攻击是在 2016 年 9 月, 相继出现了针对计算机安全撰稿人 Krebs 个人网站、法国网站托管商 OVH 以及 Dyn 公司的网络攻击事件。针对 Krebs 的初始攻击流量达到了 600 Gbps, 这是当时有记录以来最大的一次攻击, 而这种压倒性的流量就是由数十万个受控物联网设备组成的僵尸网络产生的。

2016 年 9 月 30 日, Mirai 的源代码在 hackforums.net[37]上公开发布, 研究人员依靠此代码对 Mirai 的结构和传播方式进行分析[12-13,36], 如图 8 所示:

√ Mirai 首先会进入快速扫描阶段, 在该阶段它使用 TCP SYN 探针扫描设备的 Telnet 端口;

√ 如果扫描到这两个端口, Mirai 将尝试使用预置的凭证字典暴力登录 Telnet 端口;

√ 首次登录成功后, Mirai 会将受害者的 IP 和相关凭据发送到报告服务器;

√ 之后下载程序将根据受害设备的底层系统环境(MIPS、ARM, x86 等架构)下载相应的恶意代码;

√ 成功感染后, Mirai 会通过删除已下载的恶意代码并使用随机字母数字混淆恶意进程名来隐藏其存在, 并杀死绑定到 TCP/22 或 TCP/23 以及其他竞争性的进程, 从而获得设备的独占权;

√ 此时, 指挥与控制服务器就可以向受控设备下发攻击命令, 同时扫描新的受害设备。

Mirai 的出现标志着 DDoS 攻击活动已经进入一个新的时代——越来越多的物联网设备将成为僵尸网络的目标。

Gafgyt[39]是众多僵尸网络家族中最活跃的一族, 2014 年, 黑客组织 Lizard Squad 使用 Gafgyt 家族相继对索尼 PSN 及微软 Xbox Live 发起 DDoS 攻击。2015 年 1 月, Gafgyt 的源代码被公开, 随后许多变种开始出现(如 BASHLITE, Lizkebab, Torlus 和 Qbot)。

仅到 2016 年, 已经有 100 万台 IoT 设备被该恶意软件入侵[38]: 在受感染的 100 万台终端设备中, 有将近 95%是摄像机和 DVR, 大约 4%是家用路由器, 不到 1%是受感染的 Linux 服务器。由此可见, 与过

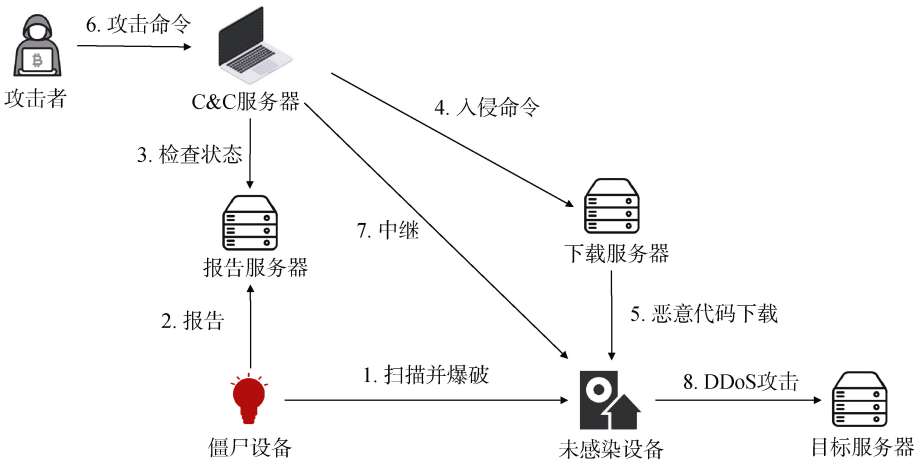


图 8　Mirai 传播流程[36]

Figure 8　Mirai propagation process[36]

去发现的基于服务器的 DDoS 僵尸网络相比, 僵尸网络的构成发生了急剧变化。

3　智能家居设备攻击与防御技术

根据智能家居应用模型及攻击向量模型, 本章将从三个攻击维度(云、管、端)来讨论针对智能家居系统的攻击方法及防御措施。

3.1　云安全

本节统计了近五年来安全领域顶级会议上有关于物联网平台安全的相关研究, 并根据其涉及的研究角度对云平台攻击与防御方法进行总结, 如表 5 所示, 具体描述如下:

表 5　云平台层面攻击与防御方法总结

Table 5　Literature summary of attack and defense methods at the cloud platform level

攻击面	已有工作	研究角度				
		平台逻辑缺陷	权限粒度过粗	语音识别漏洞	用户隐私泄露	设备诊断恢复
云平台	sp19_Zhang[51]			√		
	ccs19_Wang[44]	√				
	sec19_Zhou[40]	√				
	sec18_Kumar[49]			√		
	ndss18_Fernandes[45]		√			
	bh16_Yang[46]		√			
	sec16_Fernandes[47]		√			
	Jonh18[54]Nips18_Jia[53]			√		
	ndss17_Jia[48]		√			
	ndss19_Zhang[50]			√		
	sec19_Apthorpe[57]				√	
	ccs18_Bastys[65]				√	
	sp21_Chen[55]			√		
	Ndss18_Wang[66]					√
	sp19_Xu[68]					√
	ArXiv20_Su[56]				√	
	CCS20_Cheng[58]				√	

3.1.1　平台逻辑缺陷

随着智能家居系统的应用场景越来越多样, 云管端之间的交互变得越来越复杂, 这些依赖关系为平台带来丰富功能的同时也引入了新的攻击面, 自动化规则漏洞和实体状态转换不当是最具代表性的逻辑缺陷。

1) 自动化规则漏洞: 诸如 IFTTT [41], automate.io[42]和 CloudWork[43]一类的云平台承担了家庭自动化的任务, 随着设备的种类及规则的复杂度逐渐增加, Wang 等[44]指出自动化规则与规则之间的漏洞有可能被攻击者利用, 如条件绕过、条件阻塞、动作恢复、动作冲突、动作循环和动作重复。

缓解措施: 对于 Trigger-Action 平台中的规则间漏洞, Wang 等[44]进行了分析和概括, 并开发可以检查自动化规则中易受攻击属性的 iRuler, 其架构和工作流程如图 9 所示, iRuler 使用规则解析器从移动应用程序中提取触发规则并将其转换为规则表示, 模型构建器从规则表示、用户部署配置和设备元数据三者获得中间表示, 最后检查引擎对中间表示进行规则漏洞的扫描。iRuler 结合了可满足性模理论(Satisfiability Modulo Theories, SMT)和模型检查的功能, 为 IoT 系统建模并检查易受攻击的属性, 另外还使用自然语言处理(Natural Language Processing, NLP)进行辅助, 用于推断专有触发动作平台中规则之间的不足信息。受限制于真实规则需要大量物理设备支持, iRuler 采用人工编写规则的方式来降低成本, 因此与真实情况存在一些误差, 而且由于 Trigger-Action 平台内部的规则通常是封闭的, 并由各种第三方开发, iRuler 的通用性还有待提高。

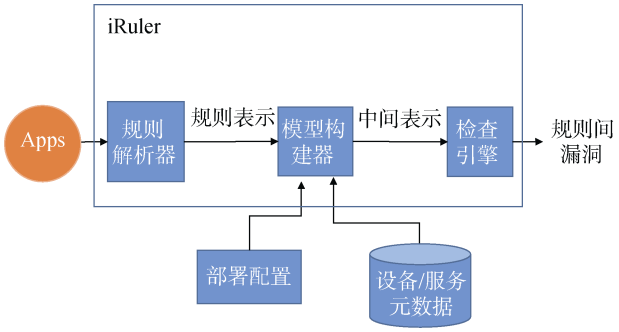
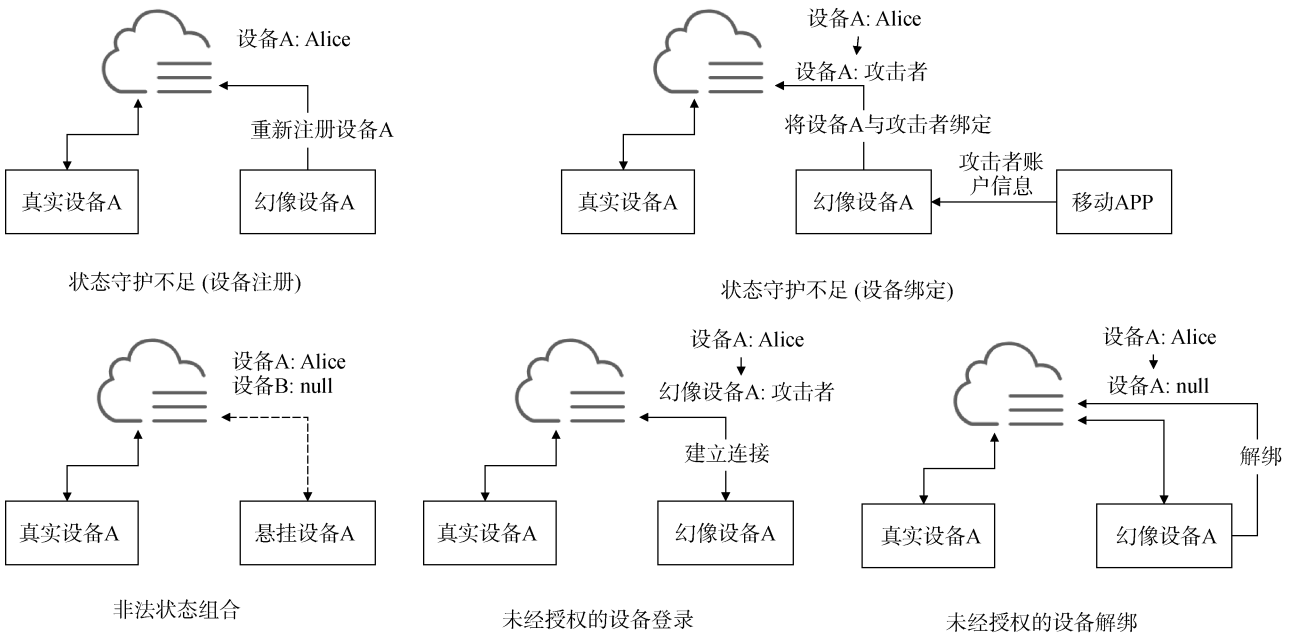


图 9 iRuler 的架构和工作流程

Figure 9 The architecture and worklow of iRuler

2) 实体状态转换不当: Zhou 等[40]对 Alink、Joylink、KASA、SmartThings 以及 MiJia 等 5 个广泛使用的智能家居平台进行了深入研究, 结合固件分析、网络流量拦截和黑盒测试, 对参与实体(即设备, 云平台和移动应用)之间的交互细节进行了逆向工程, 并重点研究了这三个实体在状态转换中的漏洞, 发现了 5 种设计缺陷, 如图 10 所示。由于大多数物联网平台允许设备在用户未授权的前提下登录和解绑, 攻击者在获得设备证书的情况下不仅可以伪装成真实设备接收用户指令, 甚至能够远程控制受害者的真实设备, 这个风险主要存在于二手交易场景中。

缓解措施: 针对实体交互中容易出现的问题, Zhou 等[40]提出要在实体交互过程中进行严格的设备认证、全面的授权检查以及有效的强化状态转换。有意思的是, 单独采用缓解措施中的某个子集是不够的, 因为交互中的缺陷是多方面共同造成的。



(注: 幻像设备指的是与真实设备使用同一证书的设备, 悬挂设备指的是未与任何用户账户绑定的设备)

图 10 实体间非法状态转换图

Figure 10 Illegal state transition diagram between entities

3.1.2 权限粒度过粗

针对平台权限控制, 安全人员需要从两个层面考虑: 一是正常权限使用者在用户不知情的情况下滥用被赋予的权限, 如任意访问用户敏感数据; 二是攻击者滥用通过远程劫持等途径获得的高级别权限, 如 OAuth 令牌泄露。

1) 敏感数据使用不当: 设备上的恶意应用程序有可能滥用用户给予的权限并泄漏数据, Fernandes 等[47]设计了安全模型 FlowFence, 它要求敏感数据的使用者明确声明预期的数据流而且强制执行声明的流, 从而有效地阻止其他非法流。

对于当前物联网平台许可模式中的设计缺陷, Jia 等[48]提出了一种适用于 IoT 平台的基于上下文的全新访问控制模型 ContexIoT, 提供对敏感操作的细粒度上下文识别, 并通过具有丰富上下文信息的运行时提示来提供上下文完整性, 以帮助用户执行有效的访问控制。与 FlowFence 不同, ContexIoT 不需要额外的开发人员工作而且满足向后兼容的需求。

2) OAuth 令牌滥用: Trigger-Action 平台通过 OAuth 令牌连接多个设备和服务以执行用户创建的自动化动作, 因此攻击者一旦掌握了 OAuth 令牌, 就可以远程操控平台上任意用户的设备[45]。虽然供应商在设计和测试云平台时下了很大工夫, 但 Yang 的报告显示[46], 在使用 OAuth 的前 600 个 Android 移动应用程序中, 有 41%容易受到远程劫持的影响。

缓解措施: 为了解决攻击者滥用 OAuth 令牌的问题, Fernandes 等[45]引入了采用分散式操作完整性(Decentralized Action Integrity)的去中心化触发动作平台(Decentralized Trigger-Action Platform, DTAP), 作为对 OAuth 协议的扩展。

3.1.3 语音识别漏洞

虚拟个人助理(Virtual Personal Assistant, VPA)平台提供的语音识别功能除了对说话者的语音内容识别之外, 还包括对说话者身份的识别, 这两种新颖的功能给使用 VPA 的智能家居带来了严重的安全问题。

1) 语音内容识别: 如图 11 VPA 平台架构所示, VPA 生态系统允许第三方人员在平台上发布类似于应用程序的技能, 与传统智能手机平台上的 APP 不同, VPA 的大多数逻辑发生在服务器上, 设备端只负责处理语音识别、录音、播放和一些基本配置。智能音箱通过准确识别说话人的语音命令来执行相应的指令操作, 但 Kumar 等人[49]发现为 Amazon Echo 系列设备提供支持的语音识别引擎 Amazon Alexa 并不能对语音命令执行绝对正确的解释, 攻击者可以通过技能抢注攻击(Skill Squatting Attack)将用户的语音命令解释为恶意应用程序。鉴于攻击者巧妙地利用一些常见的口语错误, Zhang 等[50]设计了第一个语言模型指导的模糊测试工具 LipFuzzer, 用来大规模评估意图分类器的安全性, 系统地发现潜在的易于误解的口语错误, 但贝叶斯网络非常依赖于数据集的规模和质量, 因此 LipFuzzer 的训练模型还可以进一步改进。Zhang 等[51]发现 VPA 不仅会因为发音而曲解语音命令, 还会因为缀词的使用导致语音抢注攻击(Voice Squatting Attack), 另外攻击者可以利用语音伪装攻击(Voice Masquerading Attack)在技能切换或技能终止时继续维持当前技能的控制权, 恶意的技能只会假装转交控制权给其他技能, 并继续收集用户的私密信息。

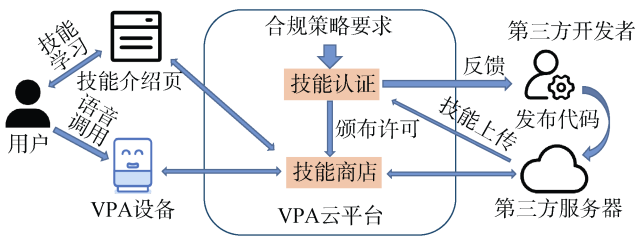


图 11 VPA 平台架构[58]

Figure 11 The architecture of VPA platforms[58]

缓解措施: 为了解决 Squatting Attack 带来的问题, Kumar 等人[49]和 Zhang 等人[51]提出基于单词和基于音素的对新技能调用名称的分析, 避免出现发音相似的技能, 同时训练一个检测器对用户意图进行判断以避免出现 Masquerading Attack。

2) 声纹识别: 除了对语音命令内容识别, 一些智能语音助手还可以支持对说话者身份识别[52], 但用户的声音却有可能成为黑客的通行证, Jia 等人[53]基于神经网络设计了语音合成系统, 可以模仿说话者的语音命令, 绕过 VPA 的认证。但这个语音合成系统生成的语音水平强烈依赖于语音数据集的数量及质量且对重音的转换效果不佳。

与 Jia 设计的欺骗攻击(Spoofing Attack)不同, Chen 等人[55]首次提出了针对声纹识别系统的黑盒对抗攻击 FAKEBOB, 对抗攻击相对于欺骗攻击的优势在于其攻击的隐蔽性更高, 它的主要设计思路是在一段语音上加入人耳无法识别的扰动音频, 从而生成对抗语音。FAKEBOB 在开源和商业声纹识别系统上达到了 99%的攻击成功率, 而且现有的四种防御手段(局部平滑、量化、音频压缩和时间依赖性检测)对 FAKEBOB 的缓解效果有限。

3.1.4 用户隐私泄露

智能家居与家庭用户的亲密关系导致智能设备能够接触并收集到消费者的隐私信息, 除了对通用隐私保护的安全研究之外, 近些年来智能音箱上 Skill 的流行也对隐私安全造成了新的威胁, 研究人员对此也进行了深入研究。

1) 通用隐私保护: 对于智能家居设备收集的用户隐私数据, 部分组织和政府推动了数据保护法规的产生, 但由于家庭用户缺乏足够的隐私意识, 供应商也没有提供透明且方便的隐私安全配置选项, 这些指导方针很难被广泛地采纳。

缓解措施: Bastys 等[65]利用访问控制和信息流控制来保护用户隐私, 并开发了一个框架来检测攻击者控制的 URL 泄漏隐私数据。Apthorpe 等[57]采用基于上下文完整性理论的调查方法, 可量化地测试这些法规所规定的条款是否确实符合目标人群预期的隐私规范。Pardis 等[63]采用三轮德尔菲法与 22 位隐私与安全专家进行意见咨询, 从而向物联网消费者提供隐私安全判断上的参考。

2) Skill 与隐私泄露: 3.1.3 集中总结了从语音、声学接口方面针对 VPA 设备技能的安全研究, 除此之外, 与 Android 生态系统中恶意应用频繁窃取用户隐私信息类似, 由于 Skill 认证过程和发布审核政策存在缺陷, VPA 平台还存在这些问题: 过度的资源访问特权[56]—通过在技能描述中提供合理的借口, 绕过权限审查; 隐蔽的后端代码更新[58,62]—Skill(技能)的后端代码运行在开发人员的服务器上, 代码在认证发布之后可以随意更新; 任意的内容篡改[56]—攻击者向新闻技能链接的网站添加不当内容。

Cheng 等人[58]对两个领先的 VPA 平台(Alexa 和谷歌)的内容及隐私策略进行了深入研究, 提交了 234 个 Alexa 技能(Skills)和 381 个谷歌动作(Actions), 这些操作均故意违反了 VPA 平台所声明的特定隐私政策, 但结果所有 Alexa Skills 和 39%的谷歌 Actions 可以通过认证, 这表明 VPA 平台在认证过程中并没有严格执行安全审核策略。

缓解措施: 为了帮助 VPA 平台提供商增强其技能认证过程的可信性, Cheng 等人[58]提出了一系列可行的缓解措施, 包括: 培训认证团队; 在技能认证过程中深入检查; 自动化技能测试工具检测违反策略的技能; 在技能生命周期中加强技能行为的完整性以缓解 VPA 后端代码更新漏洞。

从上面相关研究内容来看, 保护用户隐私简单来说需要从以下 3 方面考虑:

√ 用户需要了解设备的功能[59]以及这些功能如何影响隐私安全, 并且在使用智能家居前对可能出现的隐私安全风险充分了解。

√ 包括隐私监管机构、政府、消费者组织在内的第三方需要通过发布隐私指导及法规[60-61], 来帮助用户建立对最低隐私及安全措施的共识, 同时引导供应商在产品的整个生命周期充分考虑隐私安全问题。

√ 供应商应当向用户提供透明、方便的隐私说明及选项, 为消费者提供直观可靠的隐私提示, 参与制定并遵循第三方的安全指导, 贯彻数据/权限最小化原则。

3.1.5 设备诊断恢复

由于智能家居设备之间通过自动化规则联动, 并与家庭物理环境紧密结合, 由攻击或设备故障带来的设备异常会造成严重后果, 因此在设备遭受攻击破坏之后, 及时对设备进行诊断和恢复越来越被安全人员所重视。

1) 设备诊断: 为了能够快速定位某个设备被攻击或者配置错误的原因, ProvThings[66]以平台为中心, 对移动 APP 和设备 API 执行有效的自动化检测, 通过日志审计实现攻击调查和系统诊断。以往利用数据挖掘技术检测异常的工作存在很高的误报率和漏报率, HAWatcher[67]从智能应用程序的执行、物理交互及用户活动三个通道收集语义并相互关联, 提升了设备异常检测和诊断的效果, 但 HAWatcher 只能挖掘短时间内前后事件的相关性, 无法对长时间间隔的事件进行关联。

2) 设备恢复: 为了可以在短时间内恢复攻击者控制的设备, Xu 等[68]为设备管理者引入了恢复系统 CIDER, 管理员根据识别出的漏洞指示 CIDER 强制设备重置以清除攻击者根植在文件系统的后门, 并在设备上安装修补的固件。由于重置系统不会触及到 bootloader, 因此 CIDER 没办法恢复受损的 bootloader(如 bootkit 攻击)。

3.2 管安全

智能家居系统三大实体组件之间依靠各种通信协议交换命令和消息数据, 大致可以分为互联网协议(IP)和低功耗协议(LE)。表 6 给出了本文总结的针对这两大类通信协议的常见攻击及防御手段, 具体描述如下:

3.2.1 互联网协议安全

基于 IP 协议的应用层协议可分为 DNS、HTTP、UPnP、MQTT(消息队列遥测传输)、CoAP(受限应用协议)和私有协议。IP 协议为设备和移动端与云提供直接通信的服务, 并依靠 TCP 和 TLS/SSL 协议保证安全性。

表 6　通信管道各类协议攻击与防御方法总结

Table 6　Summary of various protocol attack and defense methods for communication pipelines

类别	协议名	已有工作	攻击方法	防御措施
IP 协议	MQTT	Samue[72]	MITM 攻击	TLS / SSL
		sp20_Jia[70]	Unauthorized MQTT Messages Faults in Managing MQTT Sessions Unauthenticated MQTT Identities Authorization Mystery of MQTT Topics	依照保护协议实体的设计原则以及采用增强的访问控制模型来强化 MQTT 的安全性
		bh18_Maggi[69]	MQTT Payload Remaining Length 正则表达式拒绝服务	保持版本更新 客户端与代理统一标准
	CoAP	bh18_Maggi[69]	UDP IP 地址欺骗和放大风险	添加防火墙规则
	DNS	Kintis [83]	ECS 破坏了 DNS 通信的私密性	禁用 ECS 功能
	HTTP	Samue[72]	中间人攻击	TLS / SSL
	UPnP	Samue[72]	中间人攻击	TLS / SSL
		Hemel[73]	NAT 规则注入	
	TLS / SSL	BEAST[76]	BEAST 攻击	保持版本更新
		CRIME[77]	CRIME 攻击	
		Lucky13[78]	Lucky Thirteen 攻击	
		POODLE[79]	POODLE 攻击	
		Bleichenbacher[81] DROWN[82]	Bleichenbacher 攻击和 DROWN 攻击	
LE 协议	BLE	bh13_Ryan[84]	TK 破解	遵循规范 BLE 4.2 的供应商和制造商应采用安全模式 1 级别 4, 遵循规范 BLE 4.0 和 BLE 4.1 的供应商和制造商应采用安全模式 1 级别 3
		Sun[88] Sivakumaran[87]	MITM 攻击	
		Zegeye[90]	LTK 暴力破解	
	ZigBee	Vidgren[91]	AVR RZ Raven USB 重放攻击 连续发送请求导致 DoS	基于信任中心(TC)的集中式安全模型
		Zillner[118]	默认信任中心链接密钥在所有设备上相同	

1) 数据协议安全: MQTT 和 CoAP 等数据协议已在 IoT 部署的 Machine-to-Machine/Man(M2M)通信中得到广泛使用, 但安全性和隐私风险值得关注。Maggi 等[69]在白皮书中表示, 暴露的 MQTT 和 CoAP 主机以及错误配置的系统可能会将凭据、敏感信息和与行业相关的过程数据泄露给潜在攻击者, 此外目前已经出现的漏洞甚至可能使攻击者获得对设备的远程控制或使其处于 DoS 状态。

由于 MQTT 协议的固有缺陷或不当使用, 主流的 IoT 云平台与设备通信消息协议的安全性很容易受到攻击, Jia 等[70-71]在对 MQTT 协议分析过程中发现存在四种攻击方式: 未经授权的 MQTT 消息(Unauthorized MQTT Messages)、管理 MQTT 会话的错误(Faults in Managing MQTT Sessions)、未经验证的 MQTT 身份标识(Unauthenticated MQTT Identities)、MQTT 主题的授权之秘(Authorization Mystery of MQTT Topics), 利用这些漏洞进行攻击会造成严重后果。

缓解措施: 针对这些漏洞, Jia等[70]提出依照保护协议实体的设计原则以及采用增强的访问控制模型来强化MQTT的安全性。

2) 基于 HTTP 的协议安全: Samue 等人[72]揭示了攻击者利用像 HTTP 这样的不安全协议在系统软件更新过程中进行中间人攻击的可能。UPnP 等协议基于 HTTP 构建, 因此不可避免地继承了 HTTP 的许多缺陷。2006 年, Armijn Hemel[73]对互联网网关设备协议(IGD)的研究表明 UPnP 容易受到各种攻击—内网计算机可能会暴露于外部网络; 2011 年, Daniel Garcia[74]在 DEFCON 19 上发布了一个工具集, 使得攻击者可以轻易利用 Hemel 发现的漏洞, 通过 WAN 接口将 NAT 规则注入到远程设备中; 2013 年, Rapid 7[75] 在 Internet 上发现了 8000 万个易受攻击的 UPnP 设备, 包括来自 1500 多家供应商的数千种型号, 未经身份验证和未加密的应用层协议等问题使攻击者能够大规模利用设备, 从而导致其他攻击, 如分布式拒绝服务攻击(DDoS)。

传输层安全性协议(TLS)及其前身安全套接层(SSL)可以为应用层的协议提供机密性和完整性的保障, 但 TLS/SSL 也并非绝对安全。 2011 年, 研究人员[76]发布了名为 BEAST(针对 SSL / TLS 的浏览器攻

击)攻击的 POC, 该攻击使中间人攻击者能够从加密的 SSL / TLS 1.0 会话中发现信息。2012 年, CRIME 攻击[77]利用 TLS 1.2 及更早版本在加密压缩数据时没有适当混淆未加密数据长度的漏洞, 攻击者可以通过观察长度差异来猜测注入内容是否匹配, 从而挖掘私密内容。2013 年, AlFardan 等人[78]提出 Lucky Thirteen 攻击, 在 MAC 验证中使用格式错误的数据包来推断时间延迟, 以从密文中统计推断明文。2014 年 10 月, Google 安全团队披露了针对 SSL 3.0 的降级漏洞 POODLE[79], 可以让攻击者破解小段的加密数据。此外, Bleichenbacher[81]攻击和 DROWN[82] 攻击进一步利用加密填充的问题说明了实现安全通信协议的困难。由于许多 IoT 通信都支持 TLS / SSL 协议的较早版本, 因此容易受到中间人(Man-in-the-middle, MITM)攻击。

缓解措施: 对于基于 HTTP 的协议, 应使用 TLS / SSL 来增加完整性和机密性。虽然 TLS/SSL 并不是绝对的安全, 但只要供应商在服务端及客户端部署最新的版本并正确配置, 就可以抵挡绝大部分攻击。

3) DNS 与隐私保护: DNS 服务使用递归查询请求的方法来完成 IP 地址与域名地址的转换, 为 Internet 的正常运作提供关键性的基础服务。Kintis 等人[83]发现 EDNS 客户端子网(edns-client-subnet, ECS)破坏了 DNS 通信的私密性: 在 ECS 下, 开放递归 DNS 会将部分隐私提供给上层机构。

缓解措施: 禁用 DNS 中的 ECS 功能可以帮助用户保护隐私安全。

3.2.2 低功耗协议安全

低功耗协议主要有 Zigbee、Z-Wave 和 Bluetooth-LE(BLE), 支持低功耗设备与家庭网关(Hub)和移动 APP 之间的通信。

1) 蓝牙协议安全: 蓝牙低功耗(Bluetooth Low Energy, BLE)是蓝牙 4.0 规范的一部份。总体而言, BLE 作为一种通信协议, 面临的主要攻击手段可以分为监听攻击和中间人攻击, 针对这两种攻击方法, BLE 使用通信加密及身份认证这两个安全机制进行防御。

Ryan[84]公开存在于 BLE4.0 及 4.1 版本中蓝牙密钥交换协议的一个严重缺陷, 这使得 BLE 通信容易受到攻击者窃听。Bluthooth SIG 在 4.2 版本[85]中引入 LE 安全连接对该漏洞进行再修补, 加上规范本身复杂性高, 增加了破解临时密钥(Temporary Key, TK)的难度。尽管 4.2 版本引入了许多确保安全要求的流程, 但制造商和销售商仍具有选择安全级别的能力, 从而可能导致各种安全事故[86]。对于蓝牙协议来说, MiTM 攻击是一个严重的安全问题, Sun 等[88]和 Sivakumaran 等[87]针对最新版本 BLE 的密钥登录配对协议进行 MITM 攻击; Jasek[89]开源了针对 BLE 的 MITM 工具—gattacker, 可以对未实现蓝牙安全功能(配对)的设备发起拒绝服务、欺骗、数据拦截、控制设备等攻击; Zegeye[90]对绑定阶段的长期密钥(Long Term Key, LTK)进行暴力破解攻击。

BLE 协议中使用 UUID 来唯一标识特定 BLE 属性(服务、特征、描述符), 从而可以建立起应用与设备之间的关系, Zuo 等人[116]开发了 BLESCOPE 对 18166 个使用了 BLE 的应用进行分析, 发现有 11141 (61.3%) 采用“直接连接(Just Works)”配对, 在没有对应用做认证的情况下, 攻击者可以随意连接到这些设备, 此外, 很多 BLE 设备采用了静态的 UUID, 攻击者可以通过嗅探 UUID 来对 BLE 设备进行指纹识别。

缓解措施: 低版本的 LE 协议普遍存在严重缺陷且攻击工具易得, 缓解方案有限, 开发者应当禁用协议中不安全的部分并保持更新。遵循 BLE 4.2 的供应商和制造商应采用安全模式 1 级别 4, 而遵循 BLE 4.0 和 BLE 4.1 的供应商和制造商应采用安全模式 1 级别 3[110]。

2) ZigBee 安全: ZigBee 技术为物联网环境中的通信提供高效且安全的短距离通信, 并尽可能降低功耗。

ZigBee 设备使用帧计数器对抗重放攻击, 但一些攻击者使用特定的软件和硬件设备绕过这一防御措施。比如 AVR RZ Raven USB[91]可以用作 ZigBee 的终端节点, 以嗅探并捕获网络流量, 并从中获得网络密钥。除使用硬件设备, Wright 还使用 python 开发了 Killerbee[92]用于捕获和分析 ZigBee 流量。DoS 是针对 ZigBee 的另一种常见攻击, 它可以损耗低功耗设备的电池寿命, Vidgren 等人[91]从理论上提出, 攻击者可以伪装成路由器或信任中心(Trust Center, TC)连续发送请求消息到终端节点, 设备必须不断响应请求, 从而对电池造成损伤。ZigBee 的协议设计也存在一些问题, Zillner 等[118]指出 ZigBee 联盟定义的默认信任中心链接密钥在所有设备上是相同的, 这让设备劫持攻击成为可能。

缓解措施: ZigBee 提供了选择各种安全模型的功能[93], 供应商应采用基于 TC 的集中式安全模型, 它被认为是最充分的安全流程。同时供应商应从 ZigBee 提供的两个安全级别(高安全性和标准安全性)中选择高安全性的。

3) Z-Wave 安全: Z-Wave 将安全性机制分为两

个主要类别: 安全性 0(S0)和安全性 2(S2)。其中 Curve25519 模型被认为是 S2 类密钥交换过程最安全的选择, 但 Genkin 等人[94]最近却对这个模型进行了侧信道攻击。到目前为止, 并未出现针对 Z-Wave 的通用性攻击方法, 但由于供应商或用户对 Z-Wave 协议的不当设置或使用, 出现了一些针对特定实现的利用。比如在 Z-Wave 与可能不包含足够安全机制的传统设备进行通信时, 攻击者可以采用重放攻击; 供应商不一定会采取 Z-Wave 提供的 AES-128 加密, Hall 等人[95]测试了 33 个 Z-Wave 设备, 发现只有 9 个支持使用加密, 并推出一个称为 EZ-Wave 的工具, 可以帮助攻击者执行各种对 Z-Wave 的渗透测试。

缓解措施: 选择使用 Z-Wave 的设备应当选择安全性更高且支持 OTA 的 Z-Wave Plus[110]。

4) 侧信道攻击: 正确地采用加密措施能够保护通信管道免受监听攻击及中间人攻击的威胁, 但一些研究人员发现事件及加密流量之间的因果关系可以用来推断智能家居中的敏感信息。PingPong[97]可以从网络流量中自动提取设备事件(例如打开/关闭灯泡)的数据包级签名特征, 攻击者可以用它来发起被动推断攻击、异常检测等。Zhang 等[98]提供了一种基于侧信道的设备行为推断技术—HoMonit, 它从 APP 源代码或 UI 交互界面提取出设备意图, 与从加密的无线通信信道(ZigBee 和 Z-wave)推断出的设备状态进行比对, 从而发现设备的异常状态。侧信道信息泄露是一把双刃剑, HoMonit 的初衷是检测行为不当的 APP, 但攻击者利用这种技术也可以发起侧信道推理攻击从而获取用户的隐私信息。由于 HoMonit 从数据包的大小和时序信息中捕获流量指纹, 因此其精度很容易受到非标准化无线流量模式的影响。此外, Luo 等[99]提出一个名为 ALTA 的应用程序级隐私泄露分析方法, 利用程序分析提取触发规则从而构建应用程序的指纹特征, 并结合从 APP 描述及输入提示中处理得到的敏感信息, 最后通过运行时动态流量分析来推断用户正在运行的应用。

缓解措施: 针对这类攻击手段, Apthorpe 等[100]提出使用流量整形保护消费者免受侧信道流量监听的威胁。

3.3 端安全-设备

终端设备承载了从物理世界采集、简单处理数据, 并根据处理结果通过执行器影响物理环境的功能。本文对终端设备攻击手段进行了总结归纳, 如表 7 所示, 常见的端侧攻击角度有固件启动、固件更新、设备传感器、过期组件。

表 7 终端设备攻击与防御方法总结

Table 7 Literature summary of terminal equipment attack and defense methods

攻击面	已有工作	研究角度				
		固件提取	过期组件	固件安全启动	固件安全更新	设备传感器
终端设备	sp17_Ronen[106]				√	
	ccs17_Zhang[107]					√
	ndss18_Roy[108]					√
	ndss20_Yan[109]					√
	sec14_Costin[105]		√			
	bh14_Hernandez[30]				√	
	sec19_Feng[111]		√			
	ccs19_Tu[91]					√
	Clinton16[101]	√				
	Barnes[102]	√				
	Bh14_Oh[112]			√		
	Bh18_Yang[104]			√		

3.3.1 固件提取

固件安全是设备安全的基石, 提取固件通常是攻击者的首要任务, 表 8 给出了本文总结的固件提取的 8 种常用方法。

Clinton 等人[101]展示了从 UART、JTAG 等硬件引脚直接访问 Echo 文件系统的难点, Barnes[102]在此基础上结合设备暴露的调试接口以及硬件配置不当(允许设备从外部 SD 卡启动), 成功获取设备 shell 并远程监听用户。随着某些供应商混淆或删除 JTAG 接口以保护其知识产权, 与 Flash 存储器直接进行交互变得非常有用(如腾讯 blade team 通过这种方法提取 Amazon Echo 固件[103])。

表 8 固件提取常用方法总结

Table 8 Summary of common methods for firmware extraction

分类	方法	攻击前提	攻击描述	防御方法
社会工程	官网或联系售后索取升级包	厂家向用户公开固件	从官网下载智能设备固件, 要求代理或官方售后提供固件	厂家不再向用户直接提供固件
软件方法	在线升级时抓包获取下载地址	设备的固件可在线升级, 固件未加密	升级固件的时候可以通过抓包的方式, 把固件给抓到	将固件内容加密
	逆向升级软件, 软件内置解包和通讯算法	上位机解密的算法可被破解	升级软件在升级前, 先在上位机解密固件, 再传输不加密的固件到设备的方式, 通过逆向上位机解密算法, 对抓取的数据包进行解密, 从而还原固件	不再通过上位机升级软件对设备升级进行管理
硬件方法	从硬件调试接口: JTAG/SWD, 利用调试工具的任意地址读取功能	在电路板上找到硬件调试接口	如果电路板上有现成的 JATG 接口, 用 JTAG 建立连接, 读出 flash 中烧录的固件	产品上线前移除硬件调试接口
	拆存储芯片, 用编程器提取固件	设备为分离式存储结构	常用的是焊下 flash 芯片, 用编程器读取固件	隐藏芯片引脚, 开启读保护
	用硬件电路的 uart 串口和固件的 BootLoader 获取固件	可以进入 BootLoader 交互界面	获取 flash 的存储信息; 用 md 命令提取固件信息; 分析输出信息, 获取固件	在编译 BootLoader 时屏蔽输入或提供有限命令
	用逻辑分析仪监听 flash、ram 获取信息	设备的 flash、ram 暴露在电路板上	把逻辑分析仪接上去就可以读取。但逻辑分析仪目前价格便宜的频率低, flash 一般频率比较高	-
	从 uart 串口获取系统权限, 打包固件	开发者提供了 uart 串口 shell	通过 uart 串口, 一些厂商为方便调试提供了系统权限, 可以通过一系列命令将文件系统打包出去	取消串口 shell

3.3.2 过期组件

在成功提取出固件之后, 攻击者通常会对固件进行解包, 然后静态逆向分析二进制程序。Costin 等人[105]完成了固件映像的首次公开大规模分析, 将 32000 个固件映像解压缩为 170 万个单独的文件, 然后对其进行静态分析, 发现了一系列漏洞。值得注意的是, 攻击者挖掘出一个 0 day 付出的成本是很高的, Feng 等 [111] 的研究揭示了近年来 IoT 攻击普遍存在的一个被忽视的原因: IoT 漏洞是公开可用且易于利用的, 而今天的 IoT 攻击几乎都是使用已知漏洞来发动恶意攻击。

缓解措施: 及时修补漏洞对于提升设备安全性有很大帮助。供应商可以通过空中下载技术(Over the Air, OTA)来修补不安全的服务和过期的组件等缺陷。

3.3.3 固件安全启动

Jeong[112]提出使用 FTDI FT2232H 芯片组进行 Bit-banging, 实现对 NAND flash 固件内容的读写, 同时, 考虑到 ECC、坏块和 JFFS2 擦除标记等问题, 他编写了自动化修改和重建固件镜像的程序。具备了重构固件的能力, 攻击者就可以通过自定义 Bootloader、内核或文件系统植入恶意代码从而持久化攻击, 目前很多厂商针对后面两者的攻击采取了一些缓解措施, 如重置按钮、内核映像校验和写入保护。YANG 等人[104]实现了名为 UbootKit 的蠕虫攻击, 该蠕虫针对 IoT 设备的引导加载程序, 可以在不同的设备之间传播, 并以 root 权限控制设备。

缓解措施: 与软件漏洞可通过 OTA 修补不同, 对于一些底层的漏洞或固有的设计缺陷, 需要在产品设计之初采用安全的框架, 例如要实现固件的安全启动, 供应商需要在片上代码中添加了完整性验证程序。

3.3.4 固件安全更新

UbootKit 攻击能够成功发起, 是因为大多数物联网设备都缺少对 Bootloader 的完整性验证, 但 UbootKit 在感染第一个僵尸设备时, 依然需要攻击者拆开设备从而暴力刷写 Flash 中的固件, 攻击途径较苛刻。不过一些智能设备缺少安全的固件更新机制, 攻击者可以通过合法的固件升级接口上传自定义固件, 例如 Nest Thermostat 固件引导过程存在后门[30], 攻击者可以通过 USB 上传恶意固件, 绕过设备对固件签名的验证; Philips Hue 智能灯对固件更新进行加密和签名, 但 Ronen 等人[106]的研究显示, Philips Hue 在灯泡之间重用对称加密和签名密钥,

攻击者能够通过边信道攻击提取主加密密钥, 并将其与通信协议中发现的漏洞结合在一起, 从而上传恶意的 OTA 更新以感染灯泡。

3.3.5 设备传感器攻击

在智能音箱设备中, 语音识别(SR)系统已经成为一种越来越流行的人机交互方法, Zhang 等人[107]设计了一个人类无法听见(频率> 20 kHz)的海豚音攻击(DolphinAttack)。如图 12 所示, 攻击者调制出超声载波上的语音命令, 音频信号通过麦克风电路的非线性特性可以成功地解调, 恢复调制前的低频音频命令, 语音识别系统对命令进行解释执行, 其中超声波只能发射 5 英尺的距离。

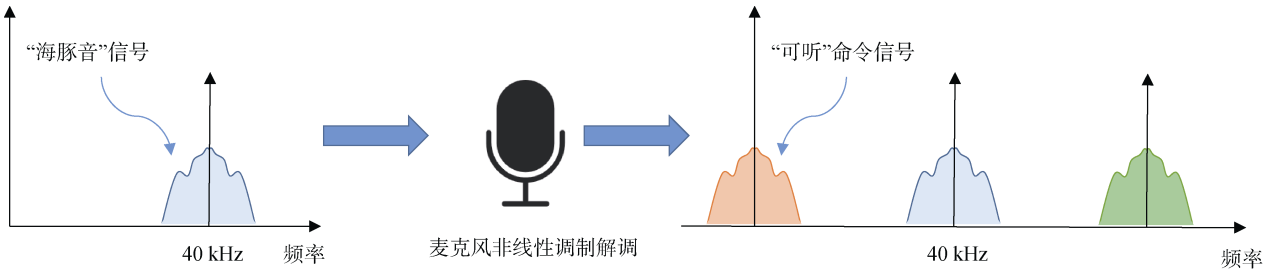


图 12 “海豚音攻击”原理图

Figure 12 Schematic diagram of “DolphinAttack”

意识到攻击范围的限制, Roy 等[108]通过聚集来自扬声器阵列的超声信号将攻击范围扩大到 25 英尺。除了空气, 声波还通过可能振动的其他材料传播。Yan 等[109]利用声音在固体介质导波传播的独特特性, 将超声中各种听不见的语音命令传递给来自不同制造商的多种目标设备, 实现了以较低的功率要求进行较长距离的攻击, 同时实现了与语音设备的多轮交互。

除了对麦克风这一传感器的攻击研究之外, Tu 等人[113]在“Trick or Heat”中利用放大器中的整流效应对温度传感器进行带外信号注入攻击, 从而控制温度传感器信号的直流电压。

传感器是智能家居的基础, 现有对传感器的安全研究表明, 除电磁干扰之外, 声音和光等不同类型的信号注入都会对传感器造成危害。由于这些攻击所利用的物理现象各不相同, 如调制解调、混叠效应和整流, 因此缓解措施没办法通用。

3.4 端安全-APP

许多智能家居设备都有配备相应的移动应用程序, 用以配置、控制和监控设备。安卓研究人员[117]使用 PlayDrone 抓取并分析了 Google Play 中上百万的应用程序, 大部分应用程序都出现了权限不当、存储不安全、编程质量不过关等问题, 智能家居相关的应用程序也不例外。表 9 给出了本文总结的针对 IoT 移动应用程序的常见攻击与防御方法。

权限使用不当: 三星的 SmartThings 是目前智能家居平台中拥有最多移动应用程序的平台, 但其安全性不容乐观, Fernandes 等[115]分析表明应用商店中超过 55%的 SmartApps 特权过高而且与 SmartApps

表 9 移动端攻击与防御方法总结

Table 9 Literature summary of mobile attack and defense methods

攻击面	已有工作	研究角度			
		权限使用不当	实体交互安全	APP 接口安全	模糊测试
移动应用终端	sp16_Fernandes[115]	√			
	Antonioli[126]			√	
	ndss19_Chen[127]				√
	Sivaraman[119]	√			
	Au[121]	√			
	Viennot[117]	√	√	√	
	ndss19_Celik[125]		√		
	sec17_Tian[120]	√			
	sec18_He[122]	√			
	Schuster[124]	√			
	Nguyen[123]		√		

异步通信的SmartThings事件子系统的安全控制不充分, 攻击者可以利用这些缺陷窃取锁定密码以及伪造假火警。此外, Sivaraman 等[119]证明了恶意移动应用程序在家庭网络会侦察家庭中的网络设备情况, 并通过端口映射将家庭内的脆弱设备或服务暴露给攻击者。

缓解措施: 为避免应用程序可以请求不必要的权限, Tian 等人[120]设计了 SmartAuth, 自动从移动 APP 的描述、代码和注释中收集与安全相关的信息, 并生成授权用户界面。Au 等人[121]开发了 PScout, 该工具可使用静态分析从 Android OS 源代码中提取权限规范, 这些规范应当被开发人员遵守来提高移动应用的安全性。He 等人[122]研究了家庭物联网的访问控制和身份验证模型的局限性并设想了基于功能的安全模型。

Schuster 等[124] 将环境形势先知(environmental situation oracles, ESO)引入物联网生态系统中作为一流的对象,设计并实施了一种新的物联网访问控制方法, 物联网访问控制框架可以使用 ESO 来强制执行情境约束。

实体交互安全: 糟糕的应用程序设计加上 APP 与设备之间的不恰当交互有可能导致整个智能家居系统进入不安全的状态。基于此, IoTSan[123]利用模型检查对智能家居中配套的移动应用程序执行静态分析, 从而预测有可能违反系统安全性能的自动化交互操作。与 IoTSan 执行的静态分析不同, Celik 等人[125]提出一种动态的、基于策略的 IoT 实施系统 IOTGUARD, 可通过监视设备和 Trigger-action 平台应用程序的行为来保障用户安全。IOTGUARD 架构如图 13 所示, 其系统包括三个组件: 代码插桩、数据采集及安全服务, 代码插桩用于收集应用程序在运行时的信息, 数据收集在程序运行时接收事件及其对应操作, 安全服务根据一系列物联网安全策略对数据收集结果进行评估, 并强制执行符合安全要求的操作。IOTGUARD 的设计及实施完全基于

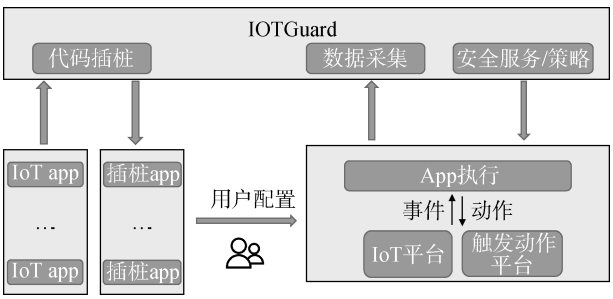


图 13 IoTGuard 系统架构

Figure 13 Architecture of the IoTGuard system

SmartThings 及 IFTTT 平台, 通用性还不够高, 另外用户与终端设备之间也存在着大量复杂交互, IOTGUARD 的监视对象还可以继续扩展。

APP 接口安全: Google 的 Nearby Connections API 帮助 Android Things 或 APP 发现附近的设备并与它们建立通信, Antonioli 等人[126]对这个闭源专有的 API 进行逆向分析, 发现并实施两类攻击: 连接操纵(connection manipulation, CMA)和范围扩展攻击(range extension attacks, REA), 对使用该 API 的所有 Android 应用程序造成威胁。

基于 APP 的模糊测试: 移动应用的分析方法相较于嵌入式设备更加成熟, 因此 Chen 等人[127]提出基于 APP 分析的黑盒 Fuzz 测试框架—IOTFUZZER, 框架利用了供应商编程到 APP 中丰富的命令(种子)消息、URL 和加密/解密信息, 避免了对设备固件提取和复杂的逆向过程。但对于消息经由云平台转发到设备的通信架构, IOTFUZZER 模糊的请求可能会被云过滤并触发防火墙警告, 从而影响进一步测试。

4 研究热点与挑战

本文统计了 2015—2020 年上半年之间, 除去综述类及调研类文章之外, 中国计算机学会推荐的网络与信息安全领域 CCF A 类和 CCF B 类英文会议与期刊中有关于智能家居安全的 90 篇文章, 对文章中重点研究的内容进行了总结概括, 并以词云的形式展现出近些年来安全人员的研究热点。

从图 14 可以看出, 隐私保护、通信加密和传感器是研究人员最为关注的话题, 这是因为智能家居配备了大量的传感器采集家庭生活中的数据, 但设备端受到计算资源的限制, 处理数据的能力有限, 这些敏感数据通常会上传到云端进行处理。如何保证传感器有效过滤掉异常输入、隐私数据在各个实体之间传输的过程不被泄露以及云平台不会滥用用户的数据, 已经成为智能家居从业人员亟需解决的问题。其次, 设备种类的增长以及家庭成员的多元化也使得身份验证及访问控制能力受到挑战, 不仅要提高安全性, 而且还要兼顾到用户在认证过程中的体验感。最后, 没有绝对安全的系统, 智能家居也不例外, 一方面需要安全人员利用模糊测试等技术构建全流程的自动化安全评估, 实现在产品上线前的安全把关; 另一方面需要供应商在设计和开发产品的过程中要考虑到设备被入侵或产生异常的情况, 建设基于云的IoT安全防御框架, 在产品上线后云平台要具备特殊情形下快速响应和恢复的能力。

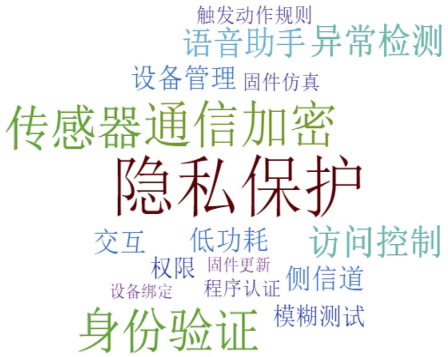


图 14 智能家居安全领域研究热点词云图

Figure 14 A word cloud of research hotspots in the field of smart home security

4.1 基于云的 IoT 安全防御

IoT 设备在本质上是易受攻击的, 首先各种 IoT 设备的功能截然不同, 更新换代的周期较长且地理分布位置广泛, 其次 IoT 设备的数量在不断增长, 许多设备的存储及计算资源有限, 无法将传统安全的一些防御措施部署到设备中, 这些因素给设备的管理者及维护者提出了安全性挑战。由于安全漏洞在不断涌现, 新的攻击向量也不断在被提出, 设备的安全性随着时间在持续衰退, 因此IoT 云服务商应该不断地监控设备运行状态及审核设备配置情况, 例如出站流量的激增可能表明设备已经被僵尸网络控制正在参与 DDoS 攻击; 为设备证书提供安全数字签名的加密算法可能随着计算机和密码学发展而削弱。如图 15 所示, 利用基于云的 IoT 安全防御机制可以及时发现设备异常行为及缺陷状态, 并通过控制台、邮件、手机短信等渠道向设备管理者告警。

IoT 安全性的基础在于控制、管理和配置设备之间的连接, 目前已有一些物联网平台向IoT开发者提供基于云的安全防御服务, 对设备的使用情况进行审核、对设备的运行状态进行实时监测。最基础的设备监控是从设备与云的通信流量中检测异常行为, 如流量大小及频率激增, 这种方式不会给设备带来额外负担, 但云平台获取的设备信息有限, 无法对设备的当前状态作出全面且精准的判断, 因此 AWS 及 Azure 这两个全球最具竞争力的物联网平台[128]向开发者提供了安装在设备上的代理, 收集物联网操作系统中的原始安全数据, 并将数据发送到云平台中心进行处理、聚合、分析、决策。本文对国内外主流的物联网平台进行了调研, 发现只有四个平台对开发者提供了设备监控服务, 其中华为和谷歌的物联网平台仅从平台日志中获取基础项目指标, 如流量指标及连接情况; AWS 及 Azure 由于在设备上安装了代理, 能够采集到更多的设备数据, 因此具有更强的风险识别能力。

表 10 基于这四个云厂商提供的安全防御能力, 对 IoT 平台网络监控及行为异常检测能力之间的异同进行了分析总结, 其中华为仅对设备消息流量的

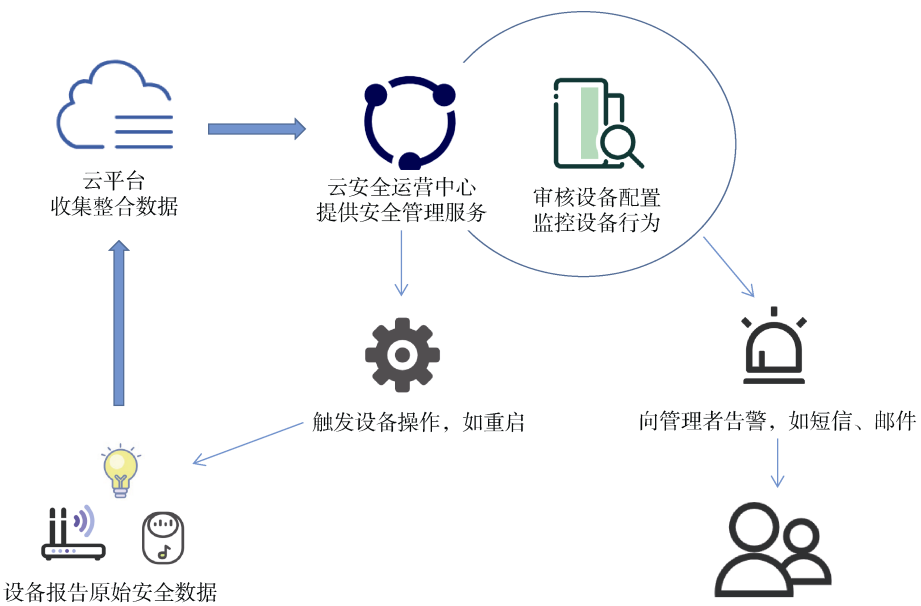


图 15 基于云的 IoT 安全防御系统原理图

Figure 15 Schematic diagram of cloud-based IoT security defense system

表 10 主流物联网平台安全防御服务检测项目

Table 10 Test items of security defense services for mainstream IoT platforms

风险类别	风险项	aws	azure	华为	谷歌
证书安全	CA 证书过期	√	-	-	-
	证书加密密钥质量问题	√	-	-	-
	CA 被吊销, 但客户端证书有效	√	-	-	-
	设备证书共享	√	-	-	-
	设备证书指纹不匹配	-	√	-	-
	设备证书过期	√	√	-	-
	撤销的设备证书仍然有效	√	-	-	-
设备认证	多设备使用相同的身份验证凭据	-	√	-	-
	设备身份验证失败	√	-	-	√
	设备暴力认证	√	√	-	-
设备连接	活跃的设备数量	-	-	-	√
	设备频繁断开连接或建连	√	-	-	√
	多个设备使用相同的客户端 ID 连接	√	-	-	-
流量异常	与不允许(黑名单)IP 进行通信	√	√	-	-
	云与设备发送字节大小	√	√	√	√
	云与设备发送报文数量	√	√	√	√
	TCP 连接数	√	-	-	-
	设备无任何遥测数据	-	√	-	-
端口异常	TCP 端口数、UDP 端口数	√	-	-	-
	监听的 TCP、UDP 端口: 80、443	√	-	-	-
	端口转发检测	-	√	-	-
	防火墙被禁用	-	√	-	-
日志痕迹	系统日志文件、Bash 历史记录被删除	-	√	-	-
	日志记录被禁用	√	√	-	-
恶意软件	类 Linux bot、勒索软件、挖矿行为	-	√	-	-
	本地登录失败次数过多	-	√	-	-
命令执行	从命令行调用/执行二进制文件	-	√	-	-
	反弹 shell、Web shell	-	√	-	-
	执行了不允许的进程	-	√	-	-
	nohup、useradd、userdel 等可疑命令	-	√	-	-
文件异常	从已知恶意软件源下载可疑文件	-	√	-	-
	设备数据丢失	-	√	-	-
	文件被篡改或替代	-	√	-	-
	可疑的编译	-	√	-	-

速度及大小进行监测, 当超过阈值时, 平台会上报告警; 谷歌使用 Cloud Monitoring API 查询和查看设备的运行指标, 除了设备流控之外, 还具备对活跃设备数量、设备连接失败次数、设备身份验证识别次数等情况进行监控的能力。亚马逊 AWS 及微软 Azure 相比前两个的基础监控能力, 利用平台的资源整合及计算能力对代理收集的设备数据进行多方位的分析, 能够对证书安全、设备认证、设备连接、流量异常、端口异常、日志痕迹等多个风险类别进行监测识别。值得一提的是, 微软的代理服务基于 C 或 C#编写, 而 AWS IoT 设备防御程序的代理 SDK 基于 Python 编写, 对计算机硬件资源和媒体文件的访问能力不如前者, 因此 Azure 具备更强大的设备状态监测能力, 能够对设备上出现的命令执行、类恶意软件及文件异常等行为持续分析和监控。

4.2 自动化漏洞挖掘技术

在通用平台上, 模糊测试在漏洞挖掘中的能力已经得到展现, 它通过向目标程序发送畸形输入, 获得程序的崩溃状态, 从而发现潜在漏洞。虽然研究人员提出了许多方法将模糊测试技术应用到嵌入式

设备上, 但由于物联网设备上的程序对硬件配置有着强烈的依赖性, 目前的研究工作依然有很大的不足和局限性, 表 11 给出了本文对固件模糊测试工具的综合比较结果, 具体描述如下:

模糊测试技术根据对目标程序的了解情况, 可以分为: 黑盒测试、灰盒测试和白盒测试。由于供应商通常不会公开智能设备的源代码, 因此针对智能设备的白盒测试相对较少。黑盒测试将目标程序视作黑盒, 依照约定的输入规则生成输入样例, 程序的执行状态无法对测试样例的生成进行指导。代表性的工具有 boofuzz[129]、Sulley[130]和 Peach[131], 这些工具需要安全人员对通信的数据格式进行复杂的分析, 而 IoTFuzzer[127] 利用了与设备配套的移动应用(带有丰富的种子、URL 及加解密信息), 可以生成更符合规则的测试用例。但这些黑盒测试工具的效率都不高, 因为无法收到目标程序的反馈, 而且真机的吞吐量很低。

表 11 IoT 固件模糊测试工具综合比较

Table 11 Comparison of IoT firmware fuzzing tools

	Avatar	Sulley	Firmadyne	IoTFuzzer	AFL	TriforceAFL	Firm-AFL
技术	白盒	黑盒	PoC	黑盒	灰盒	灰盒	灰盒
通用性	高	高	中	高	低	中	中
硬件支持	混合	无	仿真	真实	无	混合	仿真
吞吐量	非常低	低	中	低	高	中	中
0day	Y	Y	N	Y	Y	Y	Y

Muench 等人[133]的研究已经表明, 因为台式机处理器性能要远胜于物联网设备, 基于全仿真的方法实际上比真实设备要快。尽管 Chen 等人[132]提出的全系统仿真平台 Firmadyne, 相对于本地执行已经有了较大的提升, 而且 Firmadyne 解决了由于缺少实际硬件而导致的程序异常, 但测试样本的吞吐率最快也没有超过 15 个/秒。除了全系统仿真之外, 还有以 QEMU[134]为代表的用户模式仿真器以及以 Avatar[135]为代表的混合模式仿真器, 前者通过动态二进制转换提高了吞吐量, 但在目标程序有系统调用的情况下, 会因为目标程序依赖的环境与宿主机的差异而产生错误; 后者将仿真器与真实的硬件结合起来, 解决了环境依赖带来的问题, 但因为频繁的软硬件交互, 其吞吐量甚至低于真机。

仿真器只能解决黑盒测试效率不高的一部分原因。以 AFL[136]、LibFuzzer[137]、honggfuzz[138]为代表灰盒测试工具通过监控目标程序引导测试用例生成, 从而提高代码覆盖率以及测试效率。特别是 AFL, 它以提高代码覆盖率为目标, 以目标程序的执行状态为反馈, 不断丢弃无法生成新路径的输入, 并将能产生新路径的输入补充到输入队列中, 其优越性已在工业界和学术界得到广泛认同, DARPA 发起的 Cyber Grand Challenge 中的大多数决赛选手都将 AFL 用作主要的漏洞发现组件。不幸的是, AFL 通过用户模式 QEMU 支持二进制模糊测试, 因此 AFL 无法简单应用于测试 IoT 程序。为了解决 AFL 受到特定的硬件依赖性的制约, TriforceAFL[139]将 QEMU 的系统态仿真与 AFL 相结合, 实现了基于全系统仿真的模糊器。Zheng 等人[140]在此基础之上, 提出了一种新颖的技术, 即增强进程仿真 Firm-AFL, 它结合了全系统仿真的通用性和用户模式仿真的效率。但是由于其全系统仿真由 Firmadyne 支持, 因此 Firm-AFL 能够测试的固件数量取决于 Firmadyne 能正确地仿真的数量。

4.3 思考与讨论

对于自动化漏洞挖掘技术在智能家居领域的应用, 受限于硬件资源、硬件的复杂异构、代码未公开这三个因素, 具备通用性的自动化漏洞挖掘工具尚未出现, 目前在传统平台上表现出色的模糊测试工具在种子生成、固件仿真、设备监控、状态异常检测等方面还有比较大的局限性。

对于物联网平台提供的基于云的 IoT 安全防御措施, 智能家居设备的计算资源和存储能力是有限的, Azure 使用代理技术获得了良好的设备监控能力, 但这种方法对设备性能的影响是不可忽略的, 因此无法将这种技术部署到所有产品中。研究人员还需要提供统一的 IoT 威胁管理解决方案, 在无安装代理的情况下改进安全态势管理。提供全网络监控和行为异常检测能力, 需要三层威胁防护支撑: 设备配置文件审核、IoT 感知行为分析, 以及面向 IoT 的威胁情报。

设备管理者利用基于云的 IoT 安全防御可以及时发现已存在的风险, 并通过补丁和框架来缓解, 但安全性不是一个静态公式, 这要求安全运营团队能够连续建模、监视和迭代安全最佳实践, 因此需要将风险建模与验证的过程尽可能自动化。

基于此, 本文提出基于 Docker 集群的端侧自动化威胁模型, 如图 16 所示。该系统由四部分组成: 威胁建模与生成、设备管理、Docker 集群以及设备风险库。

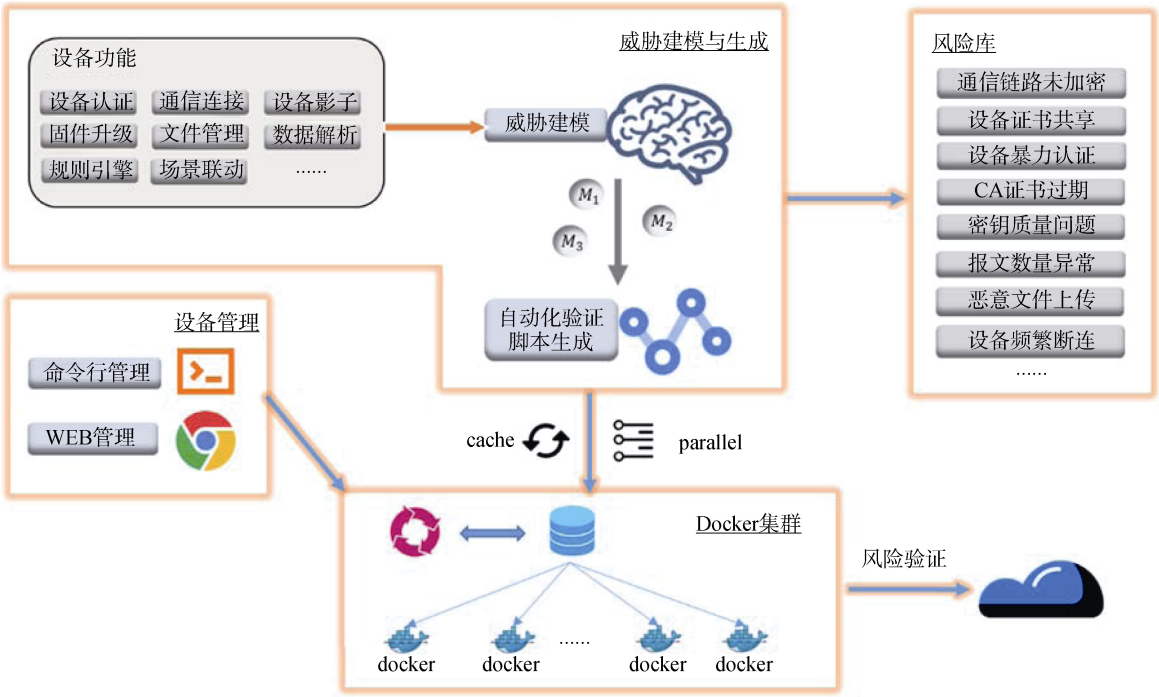


图 16　端侧自动化威胁模型架构

Figure 16　End-side automated threat model architecture

威胁建模与生成完成了从风险挖掘到自动化模板生成的过程。安全人员需要从设备功能中梳理设备的攻击面, 从中建立新的威胁模型。威胁模型经过基于模板融合的自动代码生成方法, 根据设备连接协议自动化生成威胁验证脚本。已经建立好的风险模型将存储在风险库中, 安全人员可以通过设备管理模块对所有的模型进行管理和利用。

现有的风险评估方法大多是针对传统网络系统的, 如高校网络[141-143]、办公网络[144-145]、工控系统[146-148], 这些方法大多只关心单个主机的安全性能, 而忽略了集群效应(如流量攻击), 本文提出的方案第一次将风险评估方法使用在智能家居系统中, 使用 Docker 集群来模拟设备的整个生命周期, 可以满足大规模智能家居设备模拟的需求, 能够降低真实设备运行时占用的计算资源和存储资源, 节省风险模型验证的成本。另外, 一些传统方案在真实网络环境中做渗透测试, 模拟攻击的结果直接影响线上用户, 造成用户体验不佳, 而本文的方案由于不使用真实设备也不介入线上环境, 风险验证的结果不对线上用户造成影响。

5　总结展望

智能家居包含的网络节点众多、设备种类繁杂, 使用的协议多样, 这些特点给智能家居的安全性提出了很大挑战[149]。本文在调研了关于智能家居安全的文献之后, 首先提出了一个由终端设备、云平台、移动应用程序及通信管道组成的应用模型, 然后分别从这四个层次的角度系统化阐述针对智能家居的攻击向量及缓解措施的主要研究工作, 最后对近些年学术界频繁探讨的热点议题进行了总结, 介绍了主流平台基于云的 IoT 安全防御方法, 针对模糊测试在智能家居上的应用这一技术难点进行了讨论, 并提出了基于 Docker 集群的端侧自动化威胁模型。

为维护智能家居的安全, 不仅仅需要安全研究人员的努力, 还需要以下多方共同努力, 来实现一个可信可管且安全的智能家居生态环境:

√ 供应商应对每个层面的组件进行正确的安全性开发;

√ 用户应当提高安全隐私意识并使用安全的配置策略;

√ 政府和国际组织应制定相关的法律法规;

√ 物联网标准和联盟组织应不断推动物联网安全的规范化和标准化。

结合前文对智能家居研究热点及挑战的讨论, 对于后续发展, 本文相关展望如下:

1) 用户隐私保护

智能家居的广泛应用使得家庭用户在隐私保护方面面临更大的风险, 要改善智能家居的隐私保护现状, 需要消费者、供应商及第三方(如政府)的共同努力, 如何在敏感信息的实用性和安全性之间做出合适的权衡是这三个利益相关方需要考虑的重点。

2) 访问控制策略

现有研究通常针对单个设备本身进行分析, 强调个体强壮性, 大多忽略了实体之间相互依赖的交互行为对安全的影响。由于各个实体间广泛存在的依存关系, 安全人员很难为智能家居中的各个实体划分出明确的安全边界职责, 这使得静态访问控制方法效果不佳, 因此在现有的智能家居系统中, 过度特权已成为普遍现象, 研究人员需要进一步考虑交互的多样性和平台的差异性, 从实体间的相互依赖行为入手, 设计动态的访问控制策略。

3) 固件托管

由于物联网设备的多样性, 很难为异构设备设计通用的动态分析平台, 固件托管通过对固件依赖进行建模分析, 以软件实现的方式代替硬件依赖, 但现有的固件托管工具大多仅适用于基于 Linux 的系统, 对于实时操作系统(RTOS)和裸机系统(Bare metal)的固件仿真工具还非常不成熟, 因此固件托管的支持范围还可以进一步扩展。

4) 设备异常检测和防御

由于计算资源和存储受限, 大多数物联网设备没有为系统和网络部署必要的监测和防御措施, 如何在物联网设备上利用更少的系统软件和硬件资源来实现传统安全上的防御效果, 需要安全研究人员对原有防御系统从轻量级角度进一步优化。

参考文献

[1] Manyika J, Chui M, Bisson P, et al. Unlocking the Potential of the Internet of Things. McKinsey Global Institute, 2015.

[2] Samsung, “SmartThings”, http://www.smartthings.com/,Apr 2020.

[3] Amazon, “AWS”, https://aws.amazon.com/cn/iot/, Apr 2020.

[4] Apple, “HomeKit” , http://www.apple.com/ios/homekit/, Apr 2020.

[5] Google, “Project Weave” , https://openweave.io/, Apr 2020.

[6] Microsoft, “Azure”, https://azure.microsoft.com/zh-cn/overview/iot/ , Apr 2020.

[7] Alibaba Cloud Computing, “AliyunIoT”, https://iot.aliyun.com/ ,Apr 2020.

[8] HUAWEI, “HiLink”, “http://iot.hilink.huawei.com/”, Apr 2020.

[9] XIAOMI, “mi smarthome”, https://xiaomi-mi.com/mi-smart-home/, Apr 2020.

[10] HAYLEY PETERSON, hacker breaks into smart home google nest devices terrorizes couple, https://www.businessinsider.com.au/hacker-breaks-into-smart-home-google-nest-devices-terrorizes-couple-2019-9?r=US&IR=T, Apr 2020.

[11] Stanislav M, Beardsley T. Hacking iot: A case study on baby monitor exposures and vulnerabilities. Rapid7 Report, 2015

[12] Antonakakis M, April T, Bailey M, et al. Understanding the mirai botnet[C]. *26th {USENIX} Security Symposium ({USENIX} Security 17)*. 2017: 1093-1110.

[13] Griffioen H, Doerr C. Examining Mirai's Battle over the Internet of Things[C]. The 2020 ACM SIGSAC Conference on Computer and Communications Security, 2020: 743-756.

[14] Ipvideomarket. Hikvision Backdoor Confirmed, IPVM, 18:56-400AD[EB/OL].[2019-06-25]. https://ipvm.com/reports/hik-backdoor

[15] J. Max, Backdooring the Frontdoor Hacking a “perfectly secure” smart lock. DEFCON-24, 2016.

[16] CNCERT, Internet network security situation of China in the first half of 2019[EB/OL]. https://www.cert.org.cn/publish/main/ upload/File/2019%20First%20half%20year%20.pdf,2019.8
(CNCERT, 2019 年上半年我国互联网网络安全态势[EB/OL]. https://www.cert.org.cn/publish/main/upload/File/2019%20First%20half%20year%20.pdf, 2019 年 8 月)

[17] Luo Y, Xiao Y, Cheng L, et al. Deep Learning-Based Anomaly Detection in Cyber-Physical Systems[J]. *ACM Computing Surveys*, 2021, 54(5): 1-36.

[18] Miessler D, Smith C. OWASP internet of things project. OWASP Internet of Things Project-OWASP, 2018.

[19] Kumar D, Shen K, Case B, et al. All things considered: an analysis of IoT devices on home networks[C]. *28th {USENIX} Security Symposium*, 2019: 1169-1185.

[20] Miller B A, Nixon T, Tai C, et al. Home Networking with Universal Plug and Play[J]. *IEEE Communications Magazine*, 2001, 39(12): 104-109.

[21] About Samba. https://www.samba.org/, Apr 2020.

[22] Check Point Advisories, Huawei HG532 Router Remote Code Execution, https://www.checkpoint.com/defense/advisories/public/2017/cpai-2017-1016.html/, Apr 2020.

[23] Mohurle S, Patil M. A brief study of wannacry threat: Ransomware attack 2017[J]. *International Journal of Advanced Research in Computer Science,* 2017, 8(5).

[24] Bitdefender, Ring Video Doorbell Pro Under the Scope, https://www.bitdefender.com/files/News/CaseStudies/study/294/Bitdefender-WhitePaper-RDoor-CREA3949-en-EN-GenericUse.pdf/, Apr 2020.

[25] Pierre, Pwning the Dlink 850L routers and abusing the MyDlink Cloud protocol, https://pierrekim.github.io/blog/2017-09-08-dlink-850l-mydlink-cloud-0days-vulnerabilities.html/, Apr 2020.

[26] US-CERT/NIST, CVE-2018-18778, https://nvd.nist.gov/vuln/ detail/CVE-2018-18778, 2018.

[27] Overview- Ripple20. https://www.jsof-tech.com/ripple20/, Apr 2020.

[28] Vangelis Stykas, GPS watch issues… AGAIN, https://www. pentestpartners.com/security-blog/gps-watch-issues-again, Apr 2020.

[29] Welt N. Weeping Angel: The latest surveillance tool, that can turn your smart TV into a bug TV, 2017.

[30] US-CERT/NIST, CVE-2018-20007, https://cve.mitre.org/cgi-bin/ cvename.cgi?name=CVE-2018-20007, 2018.

[31] Gatner Competitive Landscape: IoT Platform Vendors, 2020

[32] Hernandez G, Arias O, Buentello D, et al. Smart nest thermostat: A smart spy in your home. Black Hat USA, 2014 (2015).

[33] Song W N, Peng G J, Fu J M, et al. Research on Malicious Code Evolution and Traceability Technology[J]. *Journal of Software*, 2019, 30(8): 2229-2267.
(宋文纳, 彭国军, 傅建明, 等. 恶意代码演化与溯源技术研究[J]. *软件学报*, 2019, 30(8): 2229-2267.)

[34] CNCERT, Internet network security situation of China in 2019[EB/OL], https://www.cert.org.cn/publish/main/upload/File/ 2019-year.pdf
(CNCERT, 2019 年我国互联网网络安全态势[EB/OL], https:// www.cert.org.cn/publish/main/upload/File/2019-year.pdf)

[35] Kuzin M, Shmelev Y, Kuskov V. New trends in the world of IoT threats. Kaspersky Lab, 2018.

[36] Kolias C, Kambourakis G, Stavrou A, et al. DDoS in the IoT: Mirai and other Botnets[J]. *Computer*, 2017, 50(7): 80-84.

[37] Anna-senpai. [FREE] world's largest net:Mirai botnet, client,echo loader, CNC source code release. https://hackforums.net/ showthread.php?tid=5420472, Apr 2020.

[38] Black Lotus, LabsAttack Of Things! ,https://blog.centurylink. com/attack-of-things/, Apr 2020.

[39] Freebuf, Analysis of Gafgyt Family IoT Botnet Family, https:// www.freebuf.com/articles/others-articles/222677.html, Apr 2020.
(Freebuf, Gafgyt 家族物联网僵尸网络家族分析, https://www. freebuf.com/articles/others-articles/222677.html, Accessed:Apr 2020.)

[40] Zhou W, Jia Y, Yao Y, et al. Discovering and Understanding the Security Hazards in the Interactions between IoT Devices, Mobile Apps, and Clouds on Smart Home Platforms[C]. *SEC'19: Proceedings of the 28th USENIX Conference on Security Symposium*. 2019: 1133-1150.

[41] About - IFTTT, https://ifttt.com/about, Apr 2020.

[42] Work Super Smart-Automate.io, https://automate.io, Apr 2020.

[43] Cloud Business App Integration, https://cloudwork.com, Apr 2020.

[44] Wang Q, Datta P, Yang W, et al. Charting the Attack Surface of Trigger-Action IoT Platforms[C]. *The 2019 ACM SIGSAC Conference on Computer and Communications Security*, 2019: 1439-1453.

[45] Fernandes E, Rahmati A, Jung J, et al. Decentralized action integrity for trigger-action IoT platforms[C]. *2018 Network and Distributed System Security Symposium*, 2018.

[46] Yang R, Lau W C, Liu T. Signing into one billion mobile app accounts effortlessly with OAuth2.0. Black Hat Europe, 2016.

[47] Fernandes E, Paupore J, Rahmati A, et al. Flowfence: Practical data protection for emerging iot application frameworks[C]. *25th {USENIX} Security Symposium*, 2016: 531-548.

[48] Jia Y J, Chen Q A, Wang S, et al. ContexIoT: towards providing contextual integrity to appified IoT platforms[C]. *NDSS*, 2017.

[49] Kumar D, Paccagnella R, Murley P, et al. Skill squatting attacks on amazon alexa[C]. *27th {USENIX} Security Symposium*, 2018: 33-47.

[50] Zhang Y, Xu L, Mendoza A, et al. Life after Speech Recognition: Fuzzing Semantic Misinterpretation for Voice Assistant Applications[C]. *NDSS*, 2019.

[51] Zhang N, Mi X H, Feng X, et al. Dangerous Skills: Understanding and Mitigating Security Risks of Voice-Controlled Third-Party Functions on Virtual Personal Assistant Systems[C]. *2019 IEEE Symposium on Security and Privacy*, 2019: 1381-1396.

[52] Team S. Personalized Hey Siri[J]. *Apple Machine Learning Journal,* 2018, 1(9).

[53] Jia Y, Zhang Y, Weiss R J, et al. Transfer Learning from Speaker Verification to Multispeaker Text-to-Speech Synthesis[EB/OL]. 2018

[54] John Seymour and Azeem Aqil, YOUR VOICE IS MY PASSPOR0T. Black Hat USA, 2018.

[55] Chen G K, Chen S, Fan L L, et al. Who is Real Bob? Adversarial Attacks on Speaker Recognition Systems[EB/OL]. 2019

[56] Su D, Liu J Q, Zhu S C, et al. "are You Home Alone? " "yes" Disclosing Security and Privacy Vulnerabilities in Alexa Skills[EB/OL]. 2020

[57] Apthorpe N, Varghese S, Feamster N. Evaluating the Contextual Integrity of Privacy Regulation: Parents' IoT Toy Privacy Norms Versus COPPA[C]. *SEC'19: Proceedings of the 28th USENIX Conference on Security Symposium*, 2019: 123-140.

[58] Cheng L, Wilson C, Liao S, et al. Dangerous Skills Got Certified: Measuring the Trustworthiness of Skill Certification in Voice Personal Assistant Platforms[C]. *The 2020 ACM SIGSAC Conference on Computer and Communications Security*, 2020: 1699-1716.

[59] Lau J, Zimmerman B, Schaub F. Alexa, are You Listening? [J]. *Proceedings of the ACM on Human-Computer Interaction*, 2018, 2: 1-31.

[60] Fagan M, Megas K N, Scarfone K, et al. Foundational Cybersecurity Activities for IoT Device Manufacturers[R]. National Institute

of Standards and Technology, 2020.

[61] Fagan M, Yang M, Tan A, et al. Security Review of Consumer Home 16 Internet of Things (IoT) Products[EB/OL]. 2019

[62] Smart Spies: Alexa and Google Home expose users to vishing and eavesdropping. https://srlabs.de/bites/smart-spies/. Apr 2020.

[63] Emami-Naeini P, Agarwal Y, Faith Cranor L, et al. Ask the Experts: What should be on an IoT Privacy and Security Label? [J]. *2020 IEEE Symposium on Security and Privacy*, 2020: 447-464.

[64] Zhang N, Mi X H, Feng X, et al. Dangerous Skills: Understanding and Mitigating Security Risks of Voice-Controlled Third-Party Functions on Virtual Personal Assistant Systems[C]. *2019 IEEE Symposium on Security and Privacy*, 2019: 1381-1396.

[65] Bastys I, Balliu M, Sabelfeld A. If this then What? : Controlling Flows in IoT Apps[C]. *The 2018 ACM SIGSAC Conference on Computer and Communications Security*, 2018: 1102-1119.

[66] Wang Q, Hassan W U, Bates A, et al. Fear and Logging in the Internet of Things[C]. *Network and Distributed System Security Symposium,* 2018.

[67] Fu C, Zeng Q, Du X. HAWatcher: Semantics-Aware Anomaly Detection for Appified Smart Homes[C]. *30th {USENIX} Security Symposium.*

[68] Xu M, Huber M, Sun Z C, et al. Dominance as a New Trusted Computing Primitive for the Internet of Things[C]. *2019 IEEE Symposium on Security and Privacy*, 2019: 1415-1430.

[69] Maggi F, Vosseler R, Quarta D. The fragility of industrial IoT's data backbone. Trend Micro Inc., 2018.

[70] Jia Y, Xing L Y, Mao Y H, et al. Burglars' IoT Paradise: Understanding and Mitigating Security Risks of General Messaging Protocols on IoT Clouds[C]. *2020 IEEE Symposium on Security and Privacy*, 2020: 465-481.

[71] Yan Jia, Luyi Xing, et al. Sneak into Your Room: Security Holes in the Integration and Management of Messaging Protocols on Commercial IoT Clouds T. Black Hat Europe, 2019.

[72] Samuel J, Mathewson N, Cappos J, et al. Survivable key compromise in software update systems[C]. *The 17th ACM conference on Computer and communications security*, 2010: 61-72.

[73] Hemel A. Universal Plug and Play: Dead simple or simply deadly?[C]. *5th System Administration and Network Engineering Conference*, 2006, 19.

[74] Garcia D. Universal plug and play (UPnP) mapping attacks. DEFCON-19, 2011.

[75] Moore H. Security flaws in universal plug and play: Unplug. don't play[J]. *Rapid7, Ltd,* 2013, 8.

[76] US-CERT/NIST, CVE-2011-3389, https://nvd.nist.gov/vuln/detail/CVE-2011-3389 , 2011.

[77] US-CERT/NIST, CVE-2012-4929, https://nvd.nist.gov/vuln/detail/CVE-2012-4929 , 2015.

[78] Al Fardan N J, Paterson K G. Lucky Thirteen: Breaking the TLS and DTLS Record Protocols[C]. *2013 IEEE Symposium on Security and Privacy*, 2013: 526-540.

[79] Möller B, Duong T, Kotowicz K. This POODLE bites: exploiting the SSL 3.0 fallback. Security Advisory, 2014.

[80] Beurdouche B, Bhargavan K, Delignat-Lavaud A, et al. A Messy State of the Union: Taming the Composite State Machines of TLS[C]. *2015 IEEE Symposium on Security and Privacy*, 2015: 535-552.

[81] Meyer C, Somorovsky J, Weiss E, et al. Revisiting SSL/TLS implementations: New bleichenbacher side channels and attacks[C]. *23rd {USENIX} Security Symposium*, 2014: 733-748.

[82] Aviram N, Schinzel S, Somorovsky J, et al. {DROWN}: Breaking {TLS} Using SSLv2[C]. *25th {USENIX} Security Symposium*, 2016: 689-706.

[83] Kintis P, Nadji Y, Dagon D, et al. Understanding the privacy implications of ecs[C]. *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*, 2016: 343-353.

[84] Ryan M. Bluetooth Smart: The Good, The Bad, The Ugly... and The Fix. BlackHat USA, Las Vegas, USA, 2013.

[85] Bluetooth SIG. Inc.,"Bluetooth Specification Version 4.2". 2010)[2015]. Http://www.bluetooth.com.

[86] Krejčí R, Hujňák O, Švepeš M. Security Survey of the IoT Wireless Protocols[C]. *2017 25th Telecommunication Forum*, 2017: 1-4.

[87] Sivakumaran P, Blasco Alis J. A Low Energy Profile: Analysing Characteristic Security on BLE Peripherals[C]. *The Eighth ACM Conference on Data and Application Security and Privacy*, 2018: 152-154.

[88] Sun D Z, Mu Y, Susilo W. Man-in-the-Middle Attacks on Secure Simple Pairing in Bluetooth Standard $V_{5.0}$ and Its Countermeasure[J]. *Personal and Ubiquitous Computing*, 2018, 22(1): 55-67.

[89] Jasek S. Gattacking Bluetooth smart devices. Black hat USA conference. 2016.

[90] Zegeye W K. Exploiting Bluetooth low energy pairing vulnerability in telemedicine[C]. *International Foundation for Telemetering,* 2015(51).

[91] Vidgren N, Haataja K, Patiño-Andres J L, et al. Security Threats in ZigBee-Enabled Systems: Vulnerability Evaluation, Practical Experiments, Countermeasures, and Lessons Learned[C]. *2013 46th Hawaii International Conference on System Sciences*, 2013: 5132-5138.

[92] Wright J. Killerbee: practical zigbee exploitation framework[C]. *11th ToorCon conference*, 2009, 67.

[93] Alcaraz C, Lopez J. A Security Analysis for Wireless Sensor Mesh Networks in Highly Critical Systems[J]. *IEEE Transactions on Systems*, *Man*, *and Cybernetics*, *Part C*, 2010, 40(4): 419-428.

[94] Genkin D, Valenta L, Yarom Y. May the Fourth be with You: A Microarchitectural Side Channel Attack on Several Real-World Applications of Curve25519[C]. *The 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017: 845-858.

[95] Hall J, Ramsey B. Breaking bulbs briskly by bogus broadcasts. *ShmooCon*, Washington, DC, 2016.

[96] Z-Wave Alliance, Z-Wave Transport-Encapsulation Command Class Specification, http://zwavepublic.com/sites/default/files/command_class_specs_2017A/SDS13783-5Z-WaveTransport-EncapsulationCommandClassSpecification.pdf, 2017.

[97] Trimananda R, Varmarken J, Markopoulou A, et al. Packet-Level Signatures for Smart Home Devices[J]. *Signature,* 10(13): 54.

[98] Zhang W, Meng Y, Liu Y G, et al. HoMonit: Monitoring Smart Home Apps from Encrypted Traffic[C]. *The 2018 ACM SIGSAC Conference on Computer and Communications Security*, 2018: 1074-1088.

[99] Luo Y, Cheng L, Hu H X, et al. Context-Rich Privacy Leakage Analysis through Inferring Apps in Smart Home IoT[J]. *IEEE Internet of Things Journal*, 2021, 8(4): 2736-2750.

[100] Apthorpe N, Reisman D, Feamster N. Closing the Blinds: Four Strategies for Protecting Smart Home Privacy from Network Observers[EB/OL]. 2017

[101] Clinton I, Cook L, Banik S. A survey of various methods for analyzing the amazon echo. The Citadel, The Military College of South Carolina, 2016.

[102] M. Barnes, Alexa, are you listening?. MWR Labs, 2017.

[103] Li Yuxiang, Qian Wenxiang, Wu Huiyu. Breaking Smart Speaker. DEFCON-26, 2018

[104] Jingyu YANG,Chen GENG, et al. UbootKit: A Worm Attack for the Bootloader of IoT Devices, *BlackHat Asia,* 2018

[105] Costin A, Zaddach J, Francillon A, et al. A large-scale analysis of the security of embedded firmwares[C]. *23rd {USENIX} Security Symposium*, 2014: 95-110.

[106] Ronen E, Shamir A, Weingarten A O, et al. IoT Goes Nuclear: Creating a ZigBee Chain Reaction[C]. *2017 IEEE Symposium on Security and Privacy*, 2017: 195-212.

[107] Zhang G M, Yan C, Ji X Y, et al. DolphinAttack: Inaudible Voice Commands[C]. *The 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017: 103-117.

[108] Roy N, Shen S, Hassanieh H, et al. Inaudible voice commands: The long-range attack and defense[C]. *15th {USENIX} Symposium on Networked Systems Design and Implementation*, 2018: 547-560.

[109] Yan Q, Liu K, Zhou Q, et al. SurfingAttack: Interactive Hidden Attack on Voice Assistants Using Ultrasonic Guided Waves[C]. *Network and Distributed Systems Security Symposium*, 2020.

[110] Radoglou Grammatikis P I, Sarigiannidis P G, Moscholios I D. Securing the Internet of Things: Challenges, Threats and Solutions[J]. *Internet of Things*, 2019, 5: 41-70.

[111] Feng X, Liao X, Wang X F, et al. Understanding and securing device vulnerabilities through automated bug report analysis[C]. *The 28th USENIX Security Symposium*, 2019.

[112] Oh J W. Reverse engineering flash memory for fun and benefit. Blackhat US, 2014.

[113] Tu Y Z, Rampazzi S, Hao B, et al. Trick or Heat? : Manipulating Critical Temperature-Based Control Systems Using Rectification Attacks[C]. *The 2019 ACM SIGSAC Conference on Computer and Communications Security*, 2019: 2301-2315.

[114] Ding W B, Hu H X. On the Safety of IoT Device Physical Interaction Control[C]. *The 2018 ACM SIGSAC Conference on Computer and Communications Security*, 2018: 832-846.

[115] Fernandes E, Jung J, Prakash A. Security Analysis of Emerging Smart Home Applications[C]. *2016 IEEE Symposium on Security and Privacy*, 2016: 636-654.

[116] Zuo C S, Wen H H, Lin Z Q, et al. Automatic Fingerprinting of Vulnerable BLE IoT Devices with Static UUIDs from Mobile Apps[C]. *The 2019 ACM SIGSAC Conference on Computer and Communications Security*, 2019: 1469-1483.

[117] Viennot N, Garcia E, Nieh J. A Measurement Study of Google Play[C]. *The 2014 ACM international conference on Measurement and modeling of computer systems - SIGMETRICS '14*, 2014: 221-233.

[118] T. Zillner and S. Strobl, Zigbee Exploited: The good, the bad and the ugly. Blackhat US, 2015.

[119] Sivaraman V, Chan D, Earl D, et al. Smart-Phones Attacking Smart-Homes[C]. *The 9th ACM Conference on Security & Privacy in Wireless and Mobile Networks*, 2016: 195-200.

[120] Tian Y, Zhang N, Lin Y H, et al. Smartauth: User-centered authorization for the internet of things[C]. *26th {USENIX} Security Symposium*, 2017: 361-378.

[121] Au K W Y, Zhou Y F, Huang Z, et al. PScout: Analyzing the Android Permission Specification[C]. *The 2012 ACM conference on Computer and communications security - CCS '12*, 2012: 217-228.

[122] He W, Golla M, Padhi R, et al. Rethinking access control and authentication for the home internet of things (IoT)[C]. *27th {USENIX} Security Symposium*, 2018: 255-272.

[123] Nguyen D T, Song C Y, Qian Z Y, et al. IotSan: Fortifying the Safety of IoT Systems[C]. *The 14th International Conference on emerging Networking EXperiments and Technologies*, 2018: 191-203.

[124] Schuster R, Shmatikov V, Tromer E. Situational Access Control in the Internet of Things[C]. *The 2018 ACM SIGSAC Conference on*

Computer and Communications Security, 2018: 1056-1073.

[125] Celik Z B, Tan G, McDaniel P D. IoTGuard: Dynamic Enforcement of Security and Safety Policy in Commodity IoT[C]. *NDSS*, 2019.

[126] Antonioli D, Tippenhauer N O, Rasmussen K. Nearby Threats: Reversing, Analyzing, and Attacking Google's 'Nearby Connections' on Android[J]. 2019.

[127] Chen J, Diao W, Zhao Q, et al. IoTFuzzer: Discovering Memory Corruptions in IoT Through App-based Fuzzing[C]. *NDSS*, 2018.

[128] Alfonso Velosa, Emil Berthelsen, et al. Competitive Landscape: IoT Platform Vendors. Gartner Information Technology Research, 2020.

[129] Pereyda J. Boofuzz: Network Protocol Fuzzing for Humans[EB/OL]. 2017; https://github.com/jtpereyda/boofuzz.

[130] Amini P, Portnoy A. Sulley: Pure python fully automated and unattended fuzzing framework. May, 2013.

[131] Eddington M. Peach fuzzing platform. Peach Fuzzer, 2011, 34.

[132] "Firmadyne datasheet," https://cmu.app.boxcn.net/s/hnpvf1n72uccnhyfe307rc2nb9rfxmjp/folder/6601681737, Apr 2020.

[133] Muench M, Nisi D, Francillon A, et al. Avatar2: A multi-target orchestration platform[C]. *Proc. Workshop Binary Anal. Res. (Colocated NDSS Symp.)*, 2018, 18: 1-11.

[134] Bellard F. QEMU, a fast and portable dynamic translator[C]. *USENIX Annual Technical Conference,* FREENIX Track, 2005, 41: 46.

[135] Zaddach J, Bruno L, Francillon A, et al. AVATAR: A Framework to Support Dynamic Security Analysis of Embedded Systems' Firmwares[C]. *NDSS*, 2014, 14: 1-16.

[136] Zalewski M. American fuzzy lop. 2014.

[137] Serebryany K. libFuzzer–a library for coverage-guided fuzz testing. LLVM project, 2015.

[138] Swiecki R. Honggfuzz. https://github.com/google/honggfuzz, Apr 2020.

[139] NCC Group. A linux system call fuzzer using TriforceAFL, 2017.

[140] Zheng Y, Davanian A, Yin H, et al. FIRM-AFL: high-throughput greybox fuzzing of iot firmware via augmented process emulation[C]. *28th {USENIX} Security Symposium*, 2019: 1099-1114.

[141] Shi J L. Research for the Risk Assessment of University Network Security Based on Simulated Attack[J]. *Computer Engineering & Science*, 2012, 34(12): 51-55.
(史姣丽. 基于模拟攻击的高校网络安全风险评估研究[J]. *计算机工程与科学*, 2012, 34(12): 51-55.)

[142] Lu L L, Ma X. A System Model for the United Risk Assessment of Network Security Based on Mobile Agents[J]. *Computer Engineering & Science*, 2010, 32(5): 26-29.
(陆琳琳, 马鑫. 一种基于移动代理的网络安全联合风险评估系统模型[J]. *计算机工程与科学*, 2010, 32(5): 26-29.)

[143] Strutt J E, Patrick J D, Custance N D E. A Risk Assessment Methodology for Security Advisors[C]. *The Institute of Electrical and Electronics Engineers. 29th Annual 1995 International Carnahan Conference on Security Technology*, 1995: 225-229.

[144] Li Q L, He X N. Discussion on network security risk analysis method based on attack simulation [J]. *Computer Knowledge and Technology,* 2011,7(18):4324-4325, 4343.
(李全良,贺旭娜.试论基于攻击模拟的网络安全风险分析方法[J]. *电脑知识与技术,* 2011,7(18):4324-4325, 4343.)

[145] Lü H Y, Cao Y D, Shi C X. Network Security Risk Analysis Based on Simulation Attacks [J]. *Transactions of Beijing Institute of Technology*, 2008, 28(4):338-342.
(吕慧颖,曹元大,时翠霞.基于攻击模拟的网络安全风险分析方法研究[J]. *北京理工大学学报*, 2008, 28(4):338-342.)

[146] Jiang X S. Design and Implementation of Attack Test Simulation System for Industrial Control System [D]. Nanjing University of Science and Technology, 2019.
(蒋薛松. 工控系统攻击测试模拟系统设计与实现[D]. 南京理工大学, 2019.)

[147] Lu H K. *Research on Industrial Control System Vulnerability Testing and Risk Assessment*[D]. Shanghai: East China University of Science and Technology, 2014.
(卢慧康. 工业控制系统脆弱性测试与风险评估研究[D]. 上海: 华东理工大学, 2014.)

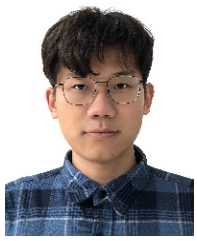
[148] Duan T, Xiang J, Zhang H, et al. Research on Attack Test Simulation Method of Industrial Control System Based on Hybrid Testing[J]. *Cyberspace Security*, 2019, 10(3): 8-22.
(段涛, 向军, 张宏, 等. 基于混合测试的工控系统攻击测试模拟方法研究[J]. *网络空间安全*, 2019, 10(3): 8-22.)

[149] Zhou W, Jia Y, Peng A N, et al. The Effect of IoT New Features on Security and Privacy: New Threats, Existing Solutions, and Challenges yet to be Solved[J]. *IEEE Internet of Things Journal*, 2019, 6(2): 1606-1616.

严寒 于 2018 年在四川大学电子信息工程专业获得学士学位。现在武汉大学网络空间安全专业攻读硕士学位。研究领域为网络安全。研究兴趣包括: IoT 安全、漏洞自动化挖掘与利用。Email: cool.yim@whu.edu.cn

彭国军 于 2008 年在武汉大学信息安全专业获得博士学位。现任武汉大学国家网络安全学院教授。研究领域为网络与信息系统安全。Email: guojpeng@whu.edu.cn

罗元 于 2015 年在武汉大学计算机学院获得学士学位。现在武汉大学国家网络安全学院网络空间安全攻读博士学位。研究领域为信息物理系统安全和移动系统安全。研究方向包括: 恶意代码检测。Email: leonnewton@whu.edu.cn



刘思德 于 2019 年在武汉大学国家网络安全学院获得学士学位。现在武汉大学国家网络安全学院攻读硕士学位。研究领域为信息安全。研究兴趣包括: 恶意代码检测与溯源、二进制逆向。Email: sidelau@whu.edu.cn