终端生态系统的

物联网安全指南

物联网服务生态系统的物联网安全指南

版本 2.0

2017 年 10 月 31 日

本文档是 GSMA 无约束力永久参考文档

安全密级：非机密

对本文档的获取和分发限于安全密级允许的人员。本文档是协会机密信息，受到版权保护。本文档仅用于所述目的，未经协会事先书面批准，不得向安全密级允许人员以外的其他人员披露文档信息或以任何方式令其获取，无论是整个文档还是文档部分内容。

版权声明

免责声明

GSM 协会（以下简称“协会”）不做与本文档信息相关的任何陈述、保证或承诺，不接受相应的责任，对本文档信息的准确性或完整性或及时性概不负责。可能变更本文档中包含的信息，恕不另行通知。

反垄断通知

本文档中包含的信息完全遵守 GSM 协会之反垄断合规政策。

目录

1 简介

1.1 GSMA 物联网安全指南文档集介绍

本文档是 GSMA 安全指南文档集的一部分，该文档集旨在帮助发展初期的“物联网”(IoT) 行业获得对物联网安全问题的一般了解。本指南文档集不具约束性，旨在倡导发展安全物联网服务的方法，以确保在整个服务周期中执行最佳安全实践。本文档就如何应对物联网服务中常见的安全威胁及薄弱环节提出建议。

GSMA 安全指南文档集结构如下所示。建议先阅读概述文档“CLP.11 物联网安全指南概述文档”[1]，然后再阅读支持性文档 CLP.12 [2] 和 CLP.13 [3]（本文档）。

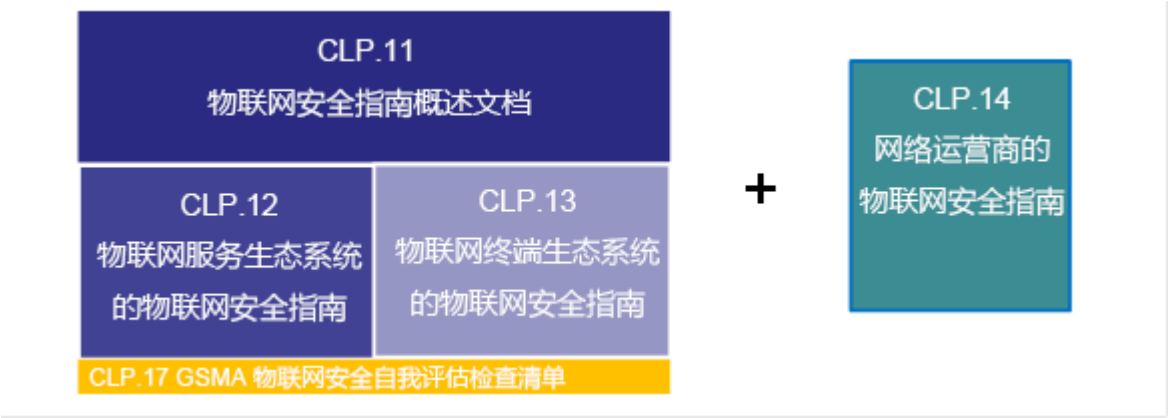

图 1 - GSMA 物联网安全指南文档结构

建议网络运营商、物联网服务供应商及物联网生态系统的其他合作伙伴阅读 GSMA 文档 CLP.14“网络运营商物联网安全指南”[4]，该文档为志在向物联网服务供应商提供服务的网络运营商提供顶级安全指南，确保系统安全和数据隐私。

1.1.1 GSMA 物联网安全评估检查清单

文档 CLP.17 [19] 中提供一份评估检查清单，借助该文档，物联网产品、服务和组件的供应商可自主评估其产品、服务和组件是否符合 GSMA 物联网安全指南。

通过完成 GSMA 物联网安全评估检查清单 [19] 中的项目，实体可以说明他们为使自己的产品、服务和组件远离网络安全风险所采取的安全措施。

完成之后可向 GSMA 呈递一份报告作为评估声明。请参阅 GSMA 网站上的相关流程：

https://www.gsma.com/iot/future-iot-networks/iot-security-guidelines/

1.2 文档目的

本文档用于从物联网终端设备角度评估物联网服务组件。从物联网的角度来看，终端是一种物理计算设备，作为连接至互联网的产品或服务的一部分执行功能或任务。例如，终端可以是可穿戴健身设备、工业控制系统、车载信息系统设备，甚至可以是个人无人机设备。用于驱动物

理设备的所有技术均应进行安全风险评估。最终产生一套实用的设计指南，让读者可以发现并修复物联网服务几乎所有的潜在风险。

本文档的范围限于与物联网终端设备设计和执行相关的建议。

本文档并非意在推动建立新物联网规格或标准，但会涉及现行解决方案、标准和最佳实践。

亦非有意加速淘汰当前的物联网服务生态。如果网络运营商现有的物联网服务足够安全，应保持对这些服务的向后兼容性。

请注意，必要时，特定地区的国家法律法规可能要高于本文档中所述的指导原则。

1.3 目标受众

本文档的主要受众包括：

- 物联网服务供应商 – 致力于开发创新互联的全新产品和服务的企业或组织。物联网服务供应商的部分运营领域包括智慧家庭、智慧城市、汽车、交通运输、健康、公共设施和消费电子产品。
- 物联网设备制造商 – 为物联网服务供应商提供物联网设备，以实现物联网服务。
- 物联网开发人员 – 代表物联网服务供应商构建物联网服务。
- 网络运营商 – 为物联网服务供应商提供服务。

1.4 定义

术语	描述
接入点名称	终端设备连接的网络连接点标识符。与服务类型相关，每个网络运营商通常配置一个接入点。
攻击者	黑客、威胁代理商、威胁执行者、诈骗者或物联网服务的其他恶意威胁。此类威胁可能来自个体犯罪、组织犯罪、恐怖主义、敌对政府及代理、工业间谍、黑客组织、政治活动分子、业余黑客、研究者以及不小心违反安全和隐私的行为。
蜂窝	任何 3GPP 标准化移动网络技术（例如 GSM、UMTS、LTE（包括 LTE-M）和 NB-IoT）。
云	互联网远程服务器网络，可担当主机、存储、管理并处理应用程序及数据。
复杂终端	终端模型可以通过长距离通信链路，如蜂窝、卫星或以太网等电路连接，持久连接后端服务器。请参考第 3 章节。
嵌入式 SIM	不易取出或更换，无法从设备中移除或更换的 SIM，可实现配置的安全变更。
终端	物联网终端是一种物理计算设备，作为连接至互联网的产品或服务的一部分执行功能或任务。有关物联网设备的三种常见类别描述和每个终端类别的示例，请参考第 3 章节。
物联网	物联网是指不同机器、设备和家用电器都可以通过不同网络连接到互联网。这些设备包括日常用品，包括平板电脑和电子消费产品、以及其他机器，如具有发送和接收数据的机对机 (M2M) 通信功能的汽车、监视器和传感器。
物联网服务	利用物联网设备数据执行服务的任何计算机程序。

术语	描述
物联网服务生态系统	服务、平台、协议及其他技术组合，可提供相关功能并从现场部署的终端中收集数据。有关详细信息，请参阅 CLP.11 [1]。
物联网服务供应商	致力于开发创新互联的新物联网产品和服务的企业或组织。
网络运营商	将物联网终端设备连接至物联网服务生态系统的通信网络运营者及所有者。
组织信任根	一系列密码政策与流程，控制如何为身份、应用程序和通信安全加密。
服务接入点	通过通信网络进入物联网服务后端基础设施的点。
用户识别模块	移动网络智能卡，用于在连接移动网络，接入网络服务时识别设备。
信任锚	在具有分层结构的密码系统中，信任锚是权威实体，信任从这里开始，且无法派生。
可信计算基	可信计算基 (TCB) 是产品或服务中算法、策略和机密的集合。通过 TCB 模块，产品或服务可以测量自身的可信度，衡量对等网络的真实性，验证产品或服务所发送或接收消息的完整性。TCB 是安全平台基础，可以在它的基础上构建安全产品及服务。TCB 的组件会根据背景（终端硬件 TCB 或云服务软件 TCB）发生变化，但抽象目标、服务、程序和策略应当非常相似。
可信执行环境 (TEE)	与富操作系统并行运行，并为该操作系统提供安全服务的环境。TEE 可采用多种技术实现，所实现的安全级别也会相应地发生改变。
UICC	ETSI TS 102 221 规定的安全元素平台，可支持以密码区分的安全域中多个标准网络或服务验证应用程序。可体现为 ETSI TS 102 671 标准中指定的嵌入式设计规格。

1.5 缩略语

术语	描述
3GPP	第 3 代项目合作伙伴
AC	交流电
API	应用程序接口
APN	接入点名称
BLE	蓝牙低功耗
BT	蓝牙
CLP	GSMA 互联生活项目
CPE	用户端设备
CPU	中央处理器
EEPROM	电子可擦除可编程只读存储器
eUICC	嵌入式 UICC
FIB	聚焦离子束
GBA	通用引导架构
GPS	全球定位系统

术语	描述
GSMA	GSM 协会
IoT	物联网
IP	互联网协议
ISM	工业、科学和医疗
LAN	局域网
LPWA	低功率广域网
LTE-M	机器长期演进
MCU	微控制器
NB-IoT	窄频带-物联网
NVRAM	非易失性随机存取存储器
OMA	开放移动联盟
PAN	个人区域网络
PSK	预共享密钥
RAM	随机存取存储器
ROM	只读存储器
SCADA	数据采集与监视控制
SPI	串行外设接口
SSH	安全外壳
SIM	用户识别模块
SRAM	静态随机存取存储器
TCB	可信计算基
TTL	晶体管-晶体管逻辑电路
UART	通用异步收发器

1.6 参考文献

参考文献	文件编号	标题
[1]	CLP.11	IoT Security Guidelines Overview Document
[2]	CLP.12	IoT Security Guidelines for IoT Service Ecosystem
[3]	CLP.13	IoT Security Guidelines for IoT Endpoint Ecosystem
[4]	CLP.14	IoT Security Guidelines for Network Operators
[5]	OMA FUMO	OMA Firmware Update Management Object www.openmobilealliance.org
[6]	不适用	ST-LINK/V2 in-circuit debugger/programmer

参考文献	文件编号	标题
		http://www.st.com/
[7]	不适用	Mobile IoT Initiative https://www.gsma.com/iot/mobile-iot-initiative/
[8]	不适用	Nmap Security Scanner https://nmap.org/
[9]	CLP.03	IoT Device Connection Efficiency Guidelines https://www.gsma.com/iot/gsma-iot-device-connection-efficiency-guidelines/
[10]	不适用	Federal Information Processing Standards www.nist.gov/itl/fips.cfmwww.nist.gov/itl/fips.cfm
[11]	不适用	EMVCo www.emvco.com/
[12]	不适用	SIM Alliance - Open Mobile API simalliance.org/key-technical-releases/
[13]	GPD_SPE_013	GlobalPlatform Secure Element Access Control www.globalplatform.org/specificationsdevice.asp
[14]	GPD_SPE_024	GlobalPlatform Trusted Execution Environment API Specification www.globalplatform.org/specificationsdevice.asp
[15]	GPC_SPE_034	GlobalPlatform Card Specification www.globalplatform.org/specificationscard.asp
[16]	ISO/IEC 29192-1	Information technology -- Security techniques -- Lightweight cryptography www.iso.org/obp/ui/#iso:std:iso-iec:29192:-1:ed-1:v1:en
[17]	TS 33.220	Generic Authentication Architecture (GAA); Generic Bootstrapping Architecture (GBA) www.3gpp.org
[18]	TS 33.222	Generic Authentication Architecture (GAA); Access to network application functions using Hypertext Transfer Protocol over Transport Layer Security (HTTPS) www.3gpp.org
[19]	CLP.17	GSMA IoT Security Assessment Checklist https://www.gsma.com/iot/iot-security-assessment/
[20]	TS -0003	oneM2M Security Solutions www.onem2m.org
[21]	3GPP TS33.163	Battery efficient Security for very low Throughput Machine Type Communication (MTC) devices (BEST)

参考文献	文件编号	标题
		www.3GPP.org

2 物联网终端的安全挑战

在很多情况下，物联网服务的安全挑战与该服务所用物联网终端的具体特性直接相关。例如，许多物联网终端的特征及相应的安全挑战如下所述：

2.1 低功耗

- 如果不可接触远程终端没有恒定电源，或者电源虽然恒定但有限（如太阳能电源），则需要使用低功耗，以延长电池寿命（通常是几年）。
- 由于更高级的加密操作功耗需求较高，低功耗终端通常只能进行简单计算的加密操作（例如终端可能仅支持 ISO/IEC 29192 [16] 中定义的轻型加密操作），并且可能仅支持有限的带宽通信，这也会使能力受限。

2.2 低成本

- 许多物联网服务的商业案例都要求将物联网终端的成本维持在较低水平。这通常会导致设备处理能力较低，存储量较少，操作系统也受到限制。最终结果就是设备可能无法执行“互联网级”加密。

2.3 使用周期长（>10 年）

- 很多终端，特别是市政和工业应用（如智能燃气表）终端，必须具有较长的使用周期。这是一项挑战，因为这意味着在设备的整个使用周期中，设备设计时所选的密码都必须保持高强度。例如，在这 10 年期间，攻击者每美元可换取的处理能力可能会增加 16 倍，而设备的能力则可能保持不变。
- 管理使用周期长的设备也是一项挑战，特别是在发现安全漏洞，但又无法在物联网终端内部修补情况下。

2.4 可实际接触

- 攻击者可以实际接触到很多物联网终端。因此，此类终端的所有硬件组件和接口都可能遭到攻击，开发者必须加以保护。

最终结果是很多物联网服务的物联网终端不会直接连接广域通信网络，而且很多物联网终端没有互联网协议 (IP) 功能。例如，物联网终端可能使用工业、科学和医疗 (ISM) 无线收发器传输数据至本地物联网服务网关，网关收到数据并通过 IP 传输至通信网络，这就使得确保端到端通信安全非常复杂。

根据物联网终端的功能及相应的安全风险，可能需要应用不同复杂程度的不同安全方法，具体内容在本文档的其他部分有所解释。

3 物联网终端模型

物联网终端模型曾被视为一组互不相干的技术，与真实世界进行交互并连接至互联网服务器以获得指导并提交指标，而今却已发生了巨大的变化。在现代工程中，物联网技术已经缩减为只有几种变体的可预测模型。物联网终端也变得越来越有预测性，其表现形式也只有以下几种：

- 轻型终端

- 复杂终端
- 网关（或“集线器”）

下图所示为一些常见的物联网终端配置：

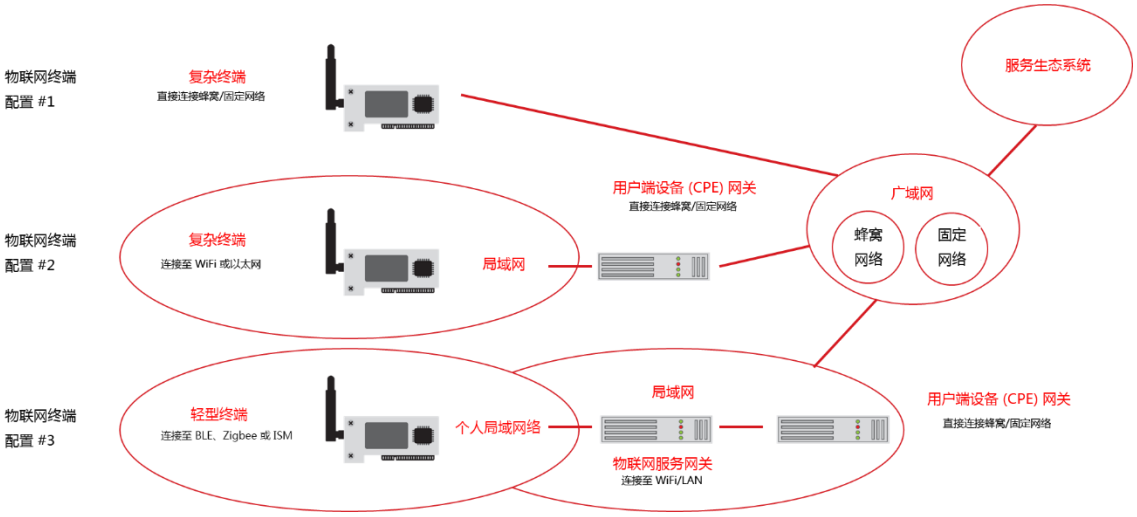

图 2 – 物联网终端配置示例

3.1 轻型终端

此类终端通常是传感器或简单的物理设备，例如功能很少的照明开关或门锁。这种终端用于单一的物理用途，可以为服务生态系统或消费者提供指标。通常采用成本较低的处理单元（可能是八位微控制器），使用短距离个人局域网络 (PAN) 或毛细协议进行连接，例如蓝牙低功耗 (BLE)、Thread 或 Zigbee。轻型终端的功率往往较低，使用纽扣电池、太阳能电源或小型锂聚合物电池即可运行。此类设备通常通过物联网服务网关和用户端设备网关连接至服务生态系统，如图 2“终端配置示例 #3”所示。

轻型终端的示例包括：

- 可穿戴设备
- 家庭安全传感器终端
- 邻近信标
- 非蜂窝毛细设备

由于轻型终端成本较低，可供其使用的安全技术极少。安全技术需要大量电流消耗、成本或电路板空间，因此通常无法用于这些系统。但是，轻型终端仍然可以使用具有成本效益的小型信任锚，来实现稳定的安全框架。

3.2 复杂终端

此类终端模型通常通过蜂窝（包括 LPWA 网络）等长距离通信链路（见图 2 中的“示例端点配置 #1”）或通过 Wi-Fi 或以太网经由用户端设备网关（见图 2 中的“示例端点配置

#2”）持续连接至后端服务器。此类设备可能内置基本处理器，甚至八位微控制器，却能够运行更稳定的处理单元，这是因为其要么直接连接至交流（AC）电源，要么包含电池，可经常使用电池充电系统。一些复杂终端可通过毛细协议进行通信，但是需要更多功率才能有效运行本地应用程序，如流音频设备。

复杂终端的示例包括：

- 连接物联网的照明系统
- 冰箱或洗衣机等家用电器
- 工业控制系统（如 SCADA）
- 改造 OBD2 蜂窝“互联汽车”跟踪和监测设备

复杂终端可以支持更高的电流消耗，通常会采用更为强大的处理器，并且电路板上也有更多空间可供安全技术使用。因此，复杂终端可以实现更多功能。此类设备几乎可以使用任何类型的信任锚。所以，它们可以很轻易地实施个性化预共享密钥（PSK）或非对称可信计算基（TCB）模型，具体内容请参见本文档后续内容。

3.3 网关（或“集线器”）

网关设备通常连接专用电源，管理着轻型终端及其驱动后端系统之间的通信。网关用于管理长距离通信链路，例如蜂窝（包括 LPWA）、卫星、固定线路、光纤或以太网。它接受服务生态系统中后端系统发出的命令，再将其转换为轻型终端可以解析的消息。终端

虽然物联网网关的主要功能是为轻型终端传输消息，但它也能执行关键任务，例如：

- 发现设备
- 部署网络驱动
- 管理功能
- 运行时监控
- GBA 或 TLS 设置等身份验证和安全功能

从技术角度而言，网关属于终端，但不一定由最终用户管理，可能由物联网服务供应商或网络运营商管理（参见下文）。尽管如此，网关仍可以被设计为复杂终端，以便更有效地利用其将上行链路分配至局部网络中多个轻型终端的功能。

与复杂终端一样，网关具备更强大的处理能力，支持更高的电流消耗，并且电路板上通常有更多空间。这样，物联网网关就可以相对容易地实施复杂的可信计算基解决方案以及 GBA 认证客户端等技术。

由于具备这些属性，网关也可以采用多种通信技术，以在不同类型的联网设备之间传输消息。这样，即可在通常无法有效交换消息的终端之间实现通信。因此，网关在本地生态系统中起到设备汇聚点的作用，让设备可以互相通信，必要时实现网络和服务生态系统间的通信。

常见的网关类型有两种，分别是“物联网服务网关”和“用户端设备（CPE）网关”。二者的区别如下：

1. “物联网服务网关”由物联网服务供应商提供。可能由最终用户拥有，但通常由物联网服务供应商管理。此类网关通常用作将轻型终端连接至服务生态系统的集线器（通过固定/蜂窝直接连接或通过 CPE 网关连接），最终用户从物联网服务供应商购买管理服务。
2. “CPE 网关”由网络运营商提供，通常是通过蜂窝或固定网络连接至互联网的宽带路由器，可用于住宅或企业环境。在此类配置中，网关通常由网络运营商管理和配置。

3.4 整体模型

无论是对哪种类型的终端进行评估或设计，从硬件和逻辑角度来看，其子组件模型都是相似的：

- 中央处理器 (CPU) 必须执行应用代码
- CPU 必须加载/存储来自/送至永久性存储的数据和可执行代码
- CPU 必须在临时性存储中计算数据
- 必须使用可信计算基验证环境
- 设备必须与其物联网生态系统通信

值得注意的是，较之复杂终端或网关，轻型终端的存储和计算能力较低，安全能力通常也较低。

整体模型最重要的方面是，每种类型的终端设备都有一项主要任务：定义可靠、高质量、安全的平台，以执行特定应用程序。也就是说，工程团队必须确保硬件为应用程序提供*可信*的平台，才能让高质量应用程序可靠运行或与其对等体安全交互，这就类似于智能手机、云服务器和大型机等更为复杂的计算平台。

本质上而言，物联网终端会加入其他终端的网络。它们不是独立设备，执行操作时会受到监督服务的影响或参与。为了提高特定设备的可信度，减少由于安全性或可靠性漏洞导致的潜在不利因素，每台终端在设计时均须秉持这样的理念：整个物联网生态系统的*可信度始于*终端硬件的构建。

从这个角度来看，很显然，即使是最容易开发的终端也必须以可靠、高质量、安全的方式运行，因为终端将加入的网络最终将连接数百万台设备。单个终端的操作必定会对整个物联网生态系统产生影响。因此，工程师不仅要考虑给定嵌入式设备的相关物理属性，更要思考架构设计的潜在影响。工程师必须考虑到整个物联网生态系统对于安全性、可靠性和质量的需求。

4 安全模型

终端的安全性可以从组件角度进行评估。通过评估构建任何给定终端所需要的各个组件，工程师和攻击者可以进行相似的攻击，毫不费力就可以使整个系统受损。

使用上文定义的整体终端模型，可以从更高层面对所用组件进行评估。通过对每个组件进行更高层面的分析，分析师可以发现经常使用，但可能得到不恰当保护的技术。分析人员或攻击者可以根据成功所需最低级别的专业知识、设备和成本划分这些组件的优先级，然后构建攻击模型，快速评估任何给定终端是否存在安全缺陷。

在终端生态系统中，对手会根据其资源、基础设施访问权限和专业知识，对几个威胁表面进行调查。这些威胁表面包括：

- 网络通信
- 可访问网络服务
- 控制台访问
- 本地总线通信
- 芯片访问

4.1 网络通信攻击

试图攻击物联网终端的第一步，也是最简单的一步，通常与通信模型漏洞相关。分析人员将观察通信模型是否采用通信安全最佳实践。如果分析人员可以轻松截获登录凭据、通信凭证或是服务生态系统用于识别终端的其他标识符，说明已经攻陷设备。

该策略可能极其简单，也可能极其困难，原因在于分析人员或攻击者对通信信道传递的明文数据的访问。设备齐全的分析人员已经掌握拦截 BLE、802.15.4 以及其他常用协议通信的技术。观察或执行针对终端通信的中间人攻击通常只需对终端进行很少的更改，或者完全不需更改，因此对攻击者非常有利。此类攻击只需极少操作即可实现。

但是，如果通信模型采用最佳实践来实施数据的保密性和完整性措施，攻击者访问有价值的机密时的难度将呈指数增加。这就导致攻击者转而使用次简单的攻击模型。

4.2 可访问网络服务攻击

攻击物联网终端的第二步是对开放的网络服务进行评估。第一步会截获终端发出的出站消息，以确定这些消息中是否有立即可用的机密。这样，攻击者可以减少从终端本身析取机密所需的工作量。如果出站通信安全模型较为完善，就可以对网络服务进行扫描，以评估是否可以通过网络访问或控制终端操作系统。

将使用 NMap [8] 等工具进行评估，以确定网络端口是否打开。如果网络拓扑不支持 IP 功能（这在 BLE 或 IEEE 802.15.4 网络中很常见），攻击者还可以使用容易获得的工具通过相应无线协议连接至终端。

然后，攻击者会尝试向终端发送消息，确定是否可以通过操纵终端执行命令或对操作系统进行远程控制台访问。常见方法是评估安全外壳 (SSH) 或远程登录等网络登录界面是否可用。如果使用默认登录凭据，攻击者可能能够登入终端。这样攻击者就能够操纵本地操作系统，并可能利用本地漏洞升级权限并析取设备中的机密。

另一个常见例子则利用设计不完善的网络服务，可以通过公共网关接口 (CGI) 脚本注入命令，这些脚本不能充分剥离用户输入字段中的控制字符，导致代码得以在本地操作系统上执行。

4.3 控制台访问攻击

确切地说，控制台访问不是攻击，而是策略。通常需要在终端上启用控制台，这样开发人员和质量保证 (QA) 技术人员才能诊断硬件或软件异常。然而，控制台提供的信息对于攻击者而言价值极高。此外，攻击者还可以通过控制台从本地和远程登录终端系统。

通常可以通过以下方法找到终端设备上的本地硬件控制台：

- 找到电路板上表示 TTL 串行端口的 5 针接口
- 查阅 CPU 或 MCU 的规格说明，确定 UART 引脚

可以使用万用表确定 TTL 端口，因为引脚符合 TTL 典型电压规格。或者也可使用逻辑分析器猜出任何通过硬件引脚的串行数据波特率。分析人员很快即可辨别控制台是否在本地硬件上可用。

在许多情况下，分析人员只需访问控制台端口即可直接访问终端设备上的命令提示符。其他情况则需要登录凭据，但通常可以猜出。如果互联网上其他人识别出登录凭据，且所有终端登录凭据均相同，则分析人员只需在线进行 Google 搜索，查看是否有人发布了登录凭据。

可通过诊断网络协议、控制台访问协议（如 SSH 或远程登录）或其他手段获取远程控制台的访问权限。应当对这些访问方法进行评估，以确定攻击者是否可以操纵访问信道，从而获得远程控制台的访问权限。

4.4 本地总线通信攻击

如果无法通过控制台获取命令提示符，攻击者或分析人员需要开始检查硬件，以确定终端是否很容易受到攻击。攻击有多种不同形式，但会采取一些简单的步骤：

- 可写介质是否存在且可改写
- 加密信息是否通过硬件总线明文传递
- 硬件电路中能否注入影响应用程序或操作系统操作的消息，从而对攻击者有利

最简单的攻击是确定是否存在可写介质。可写介质可以是易于改写的介质，例如可写外部存储 (SD/MMC) 卡，也可以是 NVRAM 芯片或 EEPROM，它们能够通过应用程序或配置变更改写，以允许访问命令提示符或安全存储的凭证。

如果这方面保护得当，分析人员会确定加密信息是否通过硬件总线明文传递。这可能需要使用逻辑分析器拦截 EEPROM 和 CPU、微控制器和 SPI 连接的网络适配器之间的消息，或其他攻击手段。攻击可以简单快速，也可以复杂且成本高昂，具体取决于攻击的复杂程度和利用的技术。

如果攻击者无法使用上述方法拦截有价值机密，他们可能会尝试向硬件总线注入消息，以更改终端上运行的应用程序的行为。这种攻击较为困难，需要高水准的专业知识和设备，而且能够评估因应用程序而异的数据及上下文。

4.5 芯片访问攻击

如果上述攻击太过复杂或成本过高，攻击者必须转而选择更为复杂的硬件攻击手段，通常涉及到破坏芯片或是电路板上各类组件的安全，具体可能包括：

- 拆开微控制器或 CPU
- 从内部 EEPROM 或 NVRAM 析取加密信息
- 拦截内部 SRAM 消息
- 进行 X 射线分析或 FIB 逆向工程

以上所有攻击手段都需要高超的技术、电子工程相关知识和昂贵的设备。虽然大多数组织都不必担心攻击者会利用这些方法对其产品进行逆向工程处理，但这仍然是需要考虑的一项重要可能性。原因在于，如果终端设备未配置唯一的加密信息，这些攻击就只需执行 *一次*。

如果未配置唯一的加密信息，一次此类攻击就可以析取足以影响整个产品线的加密信息。这种风险十分严重，因为如果数据因为任何原因被公之于众，技术就会遭到攻击和滥用，直到补丁发布（如果能够发布的话）。因为如果数据因为任何原因向公众发布，技术将遭受攻击和滥用，直到发布补丁，如果能够发布。

5 常见安全问题

本文档根据优先级提供终端安全相关建议。但是在实际应用中，从实际角度评估建议会更有帮助。工程师通常根据技术或商业影响目标开始制定一系列建议。本部分从终端的角度概述共同目标，以及与实现该目标相关的建议。

5.1 我们如何应对克隆？

保护知识产权是现代企业的重要目标。用于构建终端产品的硬件、固件和通信技术需要投入时间、专业知识和资金，公司在打造新品牌或推出新业务时往往慎之又慎。但是，无论公司采取什么措施，总有人可以利用完全相同的硬件组件，制造出与特定产品相似的“冒牌产品”或“克隆产品”。公司对这种超出法律权限的合同和合作关系无能为力。但是可以通过一些经济有效的方法阻止对克隆产品的*使用*。

在终端通信中内置身份验证功能可以确保每台终端都有由物联网服务供应商制造的加密证明。每次后端服务或对等终端与终端设备通信，会强制终端验证其身份，以区分合法终端和*克隆*终端。如果终端无法验证其身份，对等终端或服务将拒绝与其通信。需要遵照以下建议：

- 验证终端身份
- 设计不当或未执行的相互验证

5.2 如何保护终端身份？

为了正确验证终端，工程师必须能够信任终端的加密身份。这比看上去要复杂，需要结合流程、策略和技术才能实现目标。这在*执行可信计算基*建议中会有进一步介绍，但如何在终端上编码身份验证凭证决定着整个系统的安全性。

在许多终端架构中，要想冒充设备，攻击者只需要从目标设备中复制加密令牌（如有）。如果物联网服务供应商制造的每个终端都采用同一组加密令牌，那么攻击者只需要攻击一组令牌就可以冒充任何设备。

因此，正确建立 TCB 需要遵照以下建议：

- 执行可信计算基
- 使用信任锚
- 使用防篡改信任锚
- 将 API 用于 TCB
- 使用经过验证的随机数发生器
- 使用防篡改产品外壳
- 确保信任锚发送和接收通信的保密性和完整性

5.3 如何降低信任锚攻击的影响？

此外还需要注意，设备的制造和配置方式对生产中的终端安全性具有重要影响。制造过程决定了终端是否采用密钥安全编码。实现和配置过程决定了终端与特定消费者的关联方式，以及是否能在关联之前或之后攻击设备。

- 考虑供应链安全
- 在实施前对每个终端设备进行个性化设置
- 对每个终端进行独一无二的配置
- 隐私和唯一的终端标识符

5.4 如何降低冒充终端的可能性？

克隆设备进行商业用途后，攻击者感兴趣的攻击是冒充人或*特定*设备。这可能与特定个人的攻击直接相关，也可能不直接相关。可能只是冒充设备以绕过安全控制，如启用蓝牙的数字锁。

无论采用什么原理，都可以通过使用 TCB、个性化、身份验证以及以下手段应对此类攻击：

- 完好的正向加密
- 锁定存储器的关键部分

5.5 如何禁止冒充服务或对等体的功能？

物联网网络不仅包括终端设备，还包括网络服务和对等体。终端必须通过服务的验证，但服务也要通过终端的验证。双向验证可以确保应用程序更新等关键服务不会遭到破坏，也不会进而攻击网络。

- 终端通信安全
- 完好的正向加密
- 使用经过验证的随机数发生器
- 无线应用程序更新
- 设计不当或未执行的相互验证
- 未授权的元数据收集

5.6 如何禁止篡改固件和软件？

建立信任根后，终端可以从可信任组件进行验证。这样终端就可以建立信任基，确保下一级应用程序不会被攻击者无意（如通过有故障的 NVRAM）或有意地改写。通过以下方法实现：

- 最小可行执行平台（应用程序回滚）
- 对应用程序镜像进行加密签名
- 在内部 EEPROM 外引导加载
- 锁定存储器的关键部分
- 不安全的引导加载程序
- 使用防篡改产品外壳

5.7 如何降低远程代码执行的可能性？

如果篡改物理固件或软件未获得充足的结果，攻击者可能转而开展更复杂的攻击，如针对引导加载程序或通过总线或网络接口通信的应用程序执行代码。如果网络中所有对等体均通过验证，如本章前文所述，那么攻击者若想注入恶意内容，难度会增加。但大多数设备都需要进行一些类似于公共通信的活动，以与来自其他组织的设备交互。所以可能无法充分限制数据来源。

因此，必须严格审查从远程和物理接口进入计算机系统的数据。为了降低滥用应用程序的可能性，减少应用程序*遭到*攻击后的暴露，应考虑以下措施：

- 执行存储器保护
- 使用内部存储器处理加密信息
- 无线应用程序更新
- 按照适当权限级别运行应用程序
- 在应用程序架构中执行职责分离
- 执行语言安全
- 实施操作系统级安全增强功能
- 用户界面安全
- 第三方代码审计

5.8 如何禁止对架构的未授权调试或检测？

若攻击者拥有架构相关知识，且可以使用调试工具，通常会尝试检测标准调试和诊断实用程序，以获得对系统加密信息的访问权限，或者改写或注入有利代码。如果限制攻击者这方面的能力，就可以降低快速而隐蔽攻击（消费者可能无法觉察）的可能性。

- 使用防篡改信任锚
- 日志和诊断
- 锁定存储器的关键部分
- 异常检测

- 使用防篡改产品外壳
- 禁用调试和测试技术
- 用户界面安全

5.9 如何应对边信道攻击？

如果攻击者已经用尽常见手段，他们会转而使用更加复杂的攻击，以析取设备中的加密信息。这些攻击会评估硬件的行为方式，以确定行为模式是否可等同为数值（如一或零）或特定指令。这样，分析人员就能逐渐对嵌入式系统处理的数据进行反向工程。

此外，攻击者还可能使用昂贵的分析技术析取设备中的加密信息，或者建立极微小的电路来桥接晶片中的安全层。虽然这些攻击极难应对，但实施人员仍可以通过以下措施阻拦攻击：

- 在实施前对每个终端设备进行个性化设置
- 使用内部存储器处理加密信息
- 使用防篡改产品外壳
- 基于外设的攻击导致存储器受到污染
- 实施环境锁定阈值
- 设置电量警告阈值
- 设备停用和废弃
- 阻止隐藏组件和不可信桥接
- 阻止冷启动攻击
- 应对聚焦离子束和 X 射线

5.10 如何实施安全远程管理？

作为物联网终端生命周期的关键部分，远程管理必须得到保护，确保用于管理的通道不会遭到滥用。该问题不仅与未知第三方攻击者相关，消费者群体或物联网服务供应商内部也可能出现滥用。

- 终端密码管理
- 远程终端管理
- 日志和诊断
- 完好的正向加密
- 使用专用 APN

5.11 如何检测受损终端？

如果终端正常运行，几乎无法确定硬件或固件是否已被篡改，这取决于终端架构。不过，只要检测到异常时基础设施正在跟踪、记录和报警，就可以通过异常行为检测出被攻陷的设备。可考虑以下建议：

- 异常检测
- 使用防篡改产品外壳

- 设置电量警告阈值

5.12 如何在没有后端连接的情况下安全部署设备？

后端连接有时候可能无法使用，或者没有合适的后端连接。这时要确保安全就更加困难，因为根本无法管理安全密钥、身份和动态验证机制。尽管如此，还是可以达到较好的安全水平。可考虑以下建议：

- 执行可信计算基
- 定义组织信任根
- 在实施前对每个终端设备进行个性化设置
- 完好的正向加密
- 验证终端身份
- 没有后端连接的环境

5.13 如何确保消费者隐私？

消费者隐私是一个复杂的问题，不仅需要深入分析终端技术，还有整个物联网产品或服务。必须对整个系统的每个组件进行分析，确定是否存在潜在的隐私漏洞。请参考以下建议，以便更深入地了解如何确保隐私：

- 完好的正向加密
- 终端通信安全
- 隐私管理
- 隐私和唯一终端身份
- 使用专用 APN
- 未授权的元数据收集
- 不明显的安全风险（穿墙透视）
- 合法拦截

5.14 如何在确保隐私和安保的同时确保用户安全？

涉及应用程序及其目的、应用程序的预期运行环境、消费者类型、使用的通信技术时，安全是必须思考的问题。很多时候我们似乎需要在安全和安保之间权衡取舍，但事实并非如此。相反，可能需要改变架构模型以维护安全和安保。尽量不要为了安全放弃安保，而且一定要尽可能同时确保安全和安保。虽然这是理念上的建议，但工程团队应当经常审查安全。在开始讨论物联网安全问题时，可参考以下建议：

- 安全关键分析
- 有意和无意拒绝服务
- 合法拦截
- 考虑供应链安全

5.15 哪些问题可能无法解决？

由于物理定律、成本或缺乏技术解决方案，每个系统中都存在无法解决的风险。以下列出了其中的一些问题：

- 有意和无意拒绝服务
- 阻止隐藏组件和不可信桥接
- 不明显的安全风险（穿墙透视）
- 应对聚焦离子束和 X 射线
- 考虑供应链安全
- 合法拦截

6 重要建议

在开发安全终端时，应始终遵照以下建议。以下重要建议定义了安全终端架构。如果不遵循这些建议，终端安全配置将不完善，很容易被攻击者滥用。

6.1 实现终端可信计算基

保护嵌入式系统的第一步必定是定义可信计算基 (TCB)。在终端（或类似嵌入式设备）环境中，TCB 是一系列硬件、软件和协议的组合，用以确保终端的完整性，与网络对等体进行相互验证，同时管理通信和应用程序安全。

信任锚作为 TCB 的核心，是用于存储和处理预共享密钥 (PSK) 或不对称密钥等加密信息的安全硬件技术。UICC 等信任锚不仅可用于在网络通信期间验证对等体，还可以扩充以存储对终端应用程序安全有用的数据。

选择信任锚并集成至终端解决方案后，就可以选择或设计库，将信任锚集成到整个 TCB 组合中。通过 TCB，操作系统和终端的主要应用程序可以更轻松地管理设备和网络的整体安全。

工程团队必须为解决方案选择正确的信任锚，因为信任锚和 TCB 的每种组合都对应着不同的安全级别。一些组合和信任锚实施会造成安全的假象。

可信计算基最常见的变体按照“最不安全”到“最安全”的顺序排列如下：

- 未实施（明文）
- 静态预共享密钥 (PSK)
- 静态公开密钥
- 个性化 PSK
- 个性化公开密钥

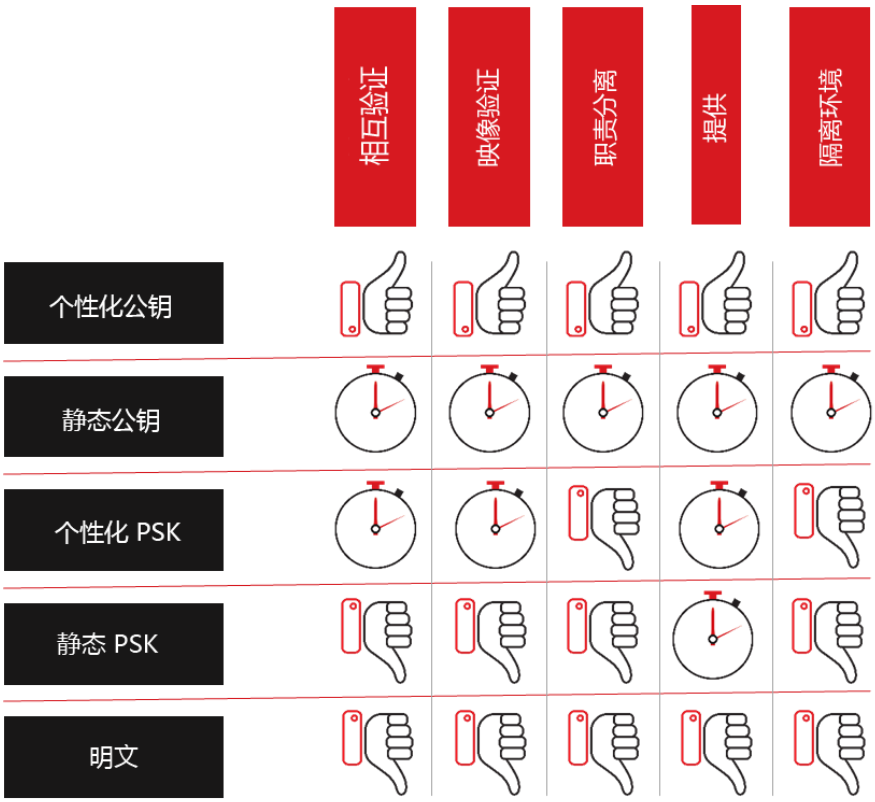

图 3 – 根据每种 TCB 提供的安全保障。

如上图所示，重点介绍了每种 TCB 变体的功能。拇指向下图标表示该 TCB 模式不能达到顶行显示的安全策略。秒表图标表示该安全策略可以使用，但在合理时间内可能遭到安全攻击。拇指向上图标表示该安全策略可以稳定实施，且安全策略的有效期很长。

虽然 TCB 可用于确保整个物联网产品和服务多个方面的安全，但本文档重点关注五个核心概念：

- 可执行映像验证
- 网络对等体相互验证
- 物联网安全架构内职责分离
- 置备和个性化
- 隔离环境安全（或无连接站点安全）

实施*可执行映像验证*的 TCB 通过对设备将要加载和执行的每个可执行映像进行密码验证，确保终端设备安全。此过程从引导加载程序开始，它会对下一个执行阶段，通常是操作系统内核，进行密码验证。引导加载程序还可以验证操作系统映像或存储在 NVRAM 中的固件应用程序映像。

实施*网络对等体相互验证*的 TCB 可以为网络组件验证提供信任根，并通过密码验证向网络对等体验证其身份。这可以提高网络对等体真正符合其声称身份的可能性。例如，如果网络对等

体声称提供固件更新服务，则 TCB 会在接受来自对等体的固件更新之前，将对等体验证为核心物联网服务供应商网络的一部分。

实施*职责分离*的 TCB 使用密钥层次结构标识物联网服务供应商提供的不同组件或服务。例如，一组密钥可以表示固件更新服务，另一组密钥可以表示“推送”服务。这些服务的功能完全不同，因此不应使用相同密钥和身份进行通信。因此，TCB 应当对每个身份进行管理和验证，以区分各个服务或功能。这样可以阻挡攻击者在攻陷其中一*个*密钥之后攻陷整个物联网服务基础设施。也就是说，攻击者攻陷“推送服务”的密钥之后，不会同时获得冒充固件更新服务的能力。

实施*个性化和置备*的 TCB 可确保终端的身份加密与该类型任何其他终端都不相同，还能保护所有通信身份，降低隐私泄露或跟踪的可能性。

实施*隔离环境安全*的 TCB 执行的策略和程序可验证对等体的真实性以及数据的机密性和完整性，即使是在没有后端服务协助的情况下。换句话说，即使长时间切断与后端服务的通信，本地化的物联网生态系统仍然能够以高度的安全性运行。虽然隔离环境的完整性会随着时间的推移逐渐降低，但经过良好设计，实施*隔离环境安全*的 TCB 可以增强网络的恢复能力，延长环境的安全时间。

在这里，*个性化*表示与特定信任锚关联的一组唯一密钥。个性化过程包括生成和安装唯一密钥，将密钥与唯一芯片关联，再将此信息和相关元数据安全传送至相应授权机构。如此可以确保每个芯片具有唯一的加密身份。在这里*静态*表示所有终端使用同一组密钥。

虽然 TCB 可用于解决嵌入式系统可能出现的几乎所有安全问题，但是 TCB 必须能够解决以下几个核心问题

- 终端应用程序映像验证
- 网络验证和/或对等体验证
- 职责分离
- 置备和个性化
- 隔离环境（无连接站点）置备和通信
- 随机化

显然选择不实施 TCB 会导致安全性缺乏，但其他常见 TCB 实施中还有一些细节需要注意。如果没有注意这些细节，可能会导致重大安全漏洞。

6.1.1 信任锚密钥模型

6.1.1.1 静态密钥

静态密钥实施（无论是 PSK 还是非对称密钥）定义为所有终端使用相同加密信息解决特定问题的解决方案。虽然可能会使用不同密钥解决不同的核心问题，但各个终端使用的密钥组合仍然是相同的。

这一模型看上去似乎很安全，因为 TCB 需要解决的每个问题都可以有效解决。但是，整个解决方案的使用期限可能较长，也可能极短。攻击者有可能在极短时间内破解此类解决方案，具体取决于所选信任锚的安全性以及加密算法和密钥大小。

问题的真正原因在于，只需攻陷一次密钥，就可以攻陷所有终端系统。这会将 TCB 实施的价值一笔勾销，在终端和整个物联网架构中实施解决方案所耗费的时间和金钱都将付诸东流。因此，这种模型的 TCB 非常危险，它实际上是一个定时炸弹。

6.1.1.2 个性化密钥

无论实施 PSK 还是非对称解决方案，都必须进行个性化才能使 TCB 有效发挥作用。个性化让攻击者无法使用攻陷的信任锚破坏整个物联网生态系统的安全。如果攻击者一次只能攻陷一个终端，并且需要通过实际接触，那么大范围攻陷物联网终端将会是一个非常缓慢、昂贵和复杂的过程。这对于企业而言是重大的胜利。

由于蜂窝通信标准在过去几十年不断演进，网络运营商已经完善了用于信任锚个性化的 PSK 模型，例如 UICC。因此，物联网终端有时可以将 UICC 置备为应用程序信任锚，从而形成具有成本效益的物联网应用程序安全解决方案。在不久的将来 eUICC 可用之后，即使现场已经部署 eUICC，也可以启用该功能。

目前，个性化密钥技术是最为有效的信任锚安全解决方案。现在物联网实施的 TCB 应当基于个性化 TCB 解决方案。物联网服务供应商应与其网络运营商讨论确定是否能实施 UICC 或 SIM 作为应用程序层信任锚。

6.1.2 TCB 协议和技术

连同信任锚，TCB 必须包含协议、策略和软件库，以保障整个物联网产品或服务的安全。采用蜂窝支持的标准信任锚的一个优点是可以适配网络运营商已有的置备和个性化软件。以下技术、协议和套件可以增强 TCB 帮助终端进行网络验证的能力。

- oneM2M TS-0003 中所述 oneM2M SM UICC 应用程序
- 通用引导架构 (GBA) 3GPP TS 33.220（参见附录 A）

使用这些技术有助于加快置备和个性化的实施，因为经验丰富的工程师和安全分析人员已经对这些库和协议进行了多年的审查。但是，这些协议可能没有完全启用 TCB 以验证终端应用程序或确保终端能够正确验证消息或授权操作。TCB 必须使用其他协议来完成这些任务，如固件验证、远程更新消息验证等。

在不远的将来，eUICC 等技术将从应用程序角度增强功能，主动 UICC 启用双重使用技术，可以帮助引导终端本身同时管理网络安全。这是一项重要的增强，因为网络运营商可以代表物联网服务供应商安全地远程管理 eUICC 设备。此外，GlobalPlatform 卡片规范 [15] 中规定的机密卡内容管理功能让物联网服务生态系统中的几个角色在网络运营商允许的前提下，可以彼此独立地管理自有应用。

6.1.3 风险

选择不实施 TCB 是整个物联网架构的关键故障点。如果没有明确定义的 TCB，信任锚和核心应用之间的相互作用定义不够严谨，可能出现漏洞，受到攻击者破坏。TCB 可确保信任锚、核心应用和网络对等体之间的通信安全、可靠且最新。如果没有 TCB，就没有中心组件来管理终端的安全生命周期。

6.2 使用信任锚

为了加入生态系统，终端必须能够验证其自身平台的完整性，并且必须能够验证其对等体的身份。为此，终端需要在可信计算基中加入信任锚。

信任锚是一个能够安全存储和处理加密信息的安全硬件元件，可以是独立物理芯片，也可以是 CPU 内的安全核心。UICC 或 eUICC 设备就是安全技术用作存储验证加密信息的信任元素的一个例子。

有效使用信任元素涉及到数据的存储、验证、更新和处理。这些数据可能是机密信息，也可能是必须进行密码验证的公开信息。在两种情况下，信任锚都必须能够安全地确定消息和身份是否能够验证，并且必须能够安全地告知 TCB 所有验证或加密操作的结果。这可以帮助应用程序和 TCB 作出影响到整个终端安全性的重要决定。例如，信任锚可以帮助终端确定网络对等体是否在冒充关键资源，如补丁部署服务器。如果信任锚无法验证网络对等体，终端上的 TCB 和应用程序应选择不与此类对等体交互，并应在可能的情况下提醒用户注意欺诈网络资源。

由于组件成本降低，需求急剧增加，信任锚比以往任何时候都更加普及。这不仅包括实际信任锚技术，还包括批准用于该技术的库和接口。这使工程团队得以在很短的时间内快速搭建信任锚解决方案，并且有助于防止定制软件或执行不力的标准缩短技术的使用期限。如有可能，应使用标准来降低出现安全漏洞的可能性。

在轻型终端中实施信任锚的另一个挑战是组件的大小。如果采用外部信任锚，就需要保持组件配置文件尽可能小。当设计规格包含 UICC 等技术时，很难实现如此之小的配置文件。不过，ETSI TS 102 671 标准引入约为 6 毫米 × 5 毫米的极小封装，解决了这一难题。“MFF1”和“MFF2”增强的 UICC 智能卡设计规格可以完全访问 UICC 支持的技术，同时确保物理要求最低。采用焊接在设备上的现场置备设计规格可以进一步增强安全性，让攻击者更难以将设备身份传送到其他设备。

开发和部署信任锚所产生的费用可能包括：

- 基础技术（嵌入 CPU 或独立芯片）的成本
- 根据需要将技术集成到电路中的成本
- 设计驱动程序或是集成到操作系统和 TCB 中的成本
- 为使用信任锚而设计应用程序的成本
- 根据需要维护信任锚
    - 维护安全密钥、撤销密钥和停用身份

- 维护保护和管理密钥及元数据所需的基础设施

- 在服务端监控信任锚身份
  - 根据需要实施设备黑名单

- 集成运营商服务（如有）以监控和管理 UICC 等信任锚

6.2.1 风险

不使用信任锚会带来很多风险，但这些风险全部源自同一个基本问题：攻击者窃取整个物联网生态系统相关密钥的能力。这样做的结果是攻击者能够实现以下操作：

- 复制终端身份
- 冒充物联网服务
- 部署未经授权的补丁或更新
- 对终端软件进行未经授权的更改

随着时间的推移，这些安全漏洞可能会为企业带来代价高昂的问题，并且不止是攻击者，竞争对手也可以通过滥用基础设施来获利。

6.3 使用防篡改信任锚

一些信任锚具有额外的物理安全性，可以防御某些类型的攻击，例如 FIB、边信道攻击和噪声干扰。虽然一些攻击从成本角度几乎无法防御，例如 FIB，但信任锚可以采用现代技术制作，以增加攻击成本。一种攻击手段的成本越高，就越不可能用于随意攻击终端设备。相反，会集中攻击物有所值的目标。

在不远的将来，一些信任锚制造商计划推出通过联邦信息处理标准 (FIPS) [10]、EMVCo [11] 和通用标准认证的技术变体。工程师在开发新技术时应确定其当前设计是否能在不久的将来迁移至合规模块。

有关更多信息，请参考各项标准的最新版本，以评估您的制造商提供的功能级别。请注意，由于实施成本和复杂性所致，一些安全级别对于消费者设备几近不可能。

6.3.1 风险

不使用防篡改信任锚的风险极高。例如，如果信任锚仅仅是嵌入 NVRAM 的密钥，任何攻击者只要具备析取这些密钥所需的工具和技术，就有可能破坏整个基础设施。但是，如果这些加密信息存储在防篡改信任锚中，析取加密信息的代价将会很高，这会降低加密信息被析取的可能性，降低信任锚作为攻击目标的价值。

值得注意的是，如果信任锚实施不力，加密信息遭到析取而导致攻陷的可能性会相当高。任何攻陷都会使设计、架构、生产和实现期间投入的费用付诸东流，可能导致重大经济损失。因此，必须确保组织设计的实施得当。

6.4 将 API 用于 TCB

在 TCB 中建立信任根之后，必须使用能够有效结合 TCB 的功能和信任根的协议。API 应能确保：

- 所有签名验证均由 TCB 执行
- TCB 不会暴露任何私有密钥
- TCB 可代表应用程序执行密钥交换
- TCB 可执行解密
- TCB 可执行加密
- TCB 可执行消息签名
- TCB 可执行安全消息填充
- TCB 和应用程序之间的机密性和完整性

这些功能有助于确保 TCB 不会将关键安全资产暴露于不安全的应用程序或硬件环境。这可以通过采用统一应用这些要求的现有规范来实现。可考虑评估下列项目：

- SIM 联盟开放移动 API [12]
- GlobalPlatform 安全元素访问控制 [13]
- GlobalPlatform 可信执行环境 (TEE) API 规范 [14]
- 可信计算组 (TCG)
- oneM2M TS-0003 [20]

许多信任锚附有可作为 TCB 实施的软件库。工程师可以使用这些库的 API 与 TCB 进行交互。优先选择信任锚提供的库（如有），因为它们很可能已经过信任锚开发领域的专家审查。但是，工程团队应对本建议给出的要求列表进行评估，并确保该库可以充分解决这些问题。

此外，*只能*从终端上运行的特权应用程序访问 TCB。切勿从终端上运行的非特权或不受信任（第三方）应用程序访问 TCB 接口。所有访问都必须通过可信服务代理，由该服务对请求进行评估并在不受信任的应用程序作出可疑或围绕隐私的请求时有选择地警告用户。

实施该协议的挑战在于需要保证所有消息在数据来源和 TCB 之间均不能被篡改，反之亦然。如果一段可从应用程序调用的 EEPROM 能够代表应用程序执行这些功能，那么这样是最有效的。将 API 代码的开头部分与内部 EEPROM 隔离，并使用内部 RAM 处理消息，可以减少暴露于外部总线的关键数据。

6.4.1 风险

如果应用程序协议接口没有明确定义，使用 TCB 可能会产生意想不到的结果或是副作用。如果工程团队提前定义协议并针对逻辑和安全问题对其进行审查，就可以更加快速有效地识别以后可能导致安全问题的漏洞。因此，在协议的定义中应对包括物联网服务供应商需求的现有 API 进行评估。如果能够找到现有的成熟技术，必定会对定制解决方案有利。

6.5 定义组织信任根

组织信任根是一系列加密策略与流程，管理着身份、应用程序和通信的安全加密方式。应使用强密码，其形式可以是唯一对称密钥、证书或公开密钥。这取决于 TCB 可使用的模型、信任锚的功能以及工程团队关注的问题。

应使用对称或非对称根私有密钥来对层次结构中使用的其他密钥进行数字签名。例如，如果我们的示例组织 Example IoT Company LLC 要创建一个组织信任根，他们会在一台可信计算机上生成根密钥。该密钥代表组织根。然后他们会生成新密钥来代表每个应当具有独立安全层级的子组织。可能的例子包括：

- 代码签名密钥
- 服务器通信密钥
- 对等通信密钥
- 终端身份密钥
- 吊销主密钥

这些密钥中的每一个均应由组织根密钥进行签名。所有这些密钥及其相应签名和根密钥均应存储在 TCB 使用的信任锚中。然后，每当使用链接至特定密钥的应用程序时，该应用程序可以使用特定密钥来验证通过通信信道发送的消息。

这一模型有助于确保所有消息均通过加密层次结构进行保护。通过在特定密钥类型之间进行职责分离，可以经由同样的通信过程吊销被攻陷的密钥。

可协助部署这一方法的现有协议包括：

- 传输层安全 (TLS)；最新有效规范
- 安全外壳 (SSH2)
- 在线证书状态协议 (OCSP) IETF RFC 2560
- 通用引导架构 (GBA)（参见附录 A）3GPP TS 33.220

如果必须部署需要密钥的服务，会遇到一些困难。应当为该服务器层专门生成单独的证书或密钥对，而不是将服务器通信密钥等安全关键资产放在可通过互联网访问的 Web 服务器上。然后可以用服务器通信密钥对此证书进行签名。如此一来，任何终端都可以验证服务是否通过信任根认证，而关键组织密钥却不会暴露给攻击者。

曾经泄露的密钥可以使用吊销主密钥验证吊销，以停止使用。

毋庸置疑，组织信任根中的所有核心密钥对于基础设施的安全性都是至关重要的。这些密钥必须得到严格保护，并且只能由核心团队的可信内部成员使用。强烈建议使用经认可的硬件安全模块 (HSM) 存储、访问和使用这些密钥。

虽然在技术部署前期，HSM 往往是一笔不菲的开销，但对财务的长期影响是积极的。这样只会产生一笔相对较小的前期费用，而不是后期在鉴定分析和工程设计中花费大量费用诊断和应对本可以由 TCB 和 HSM 解决的特定风险。

6.5.1 风险

不使用组织信任根的风险是单个密钥遭到任何攻击都可能导致整个生态系统被攻陷。将组织分为层次结构，再为层次结构部署不同的密钥，就可以根据密钥关联的应用程序或子组织优先级，定期循环使用密钥。这样可以实现组织各方面职责分离，同时防止整个基础设施的安全由于单个密钥被攻陷而遭到破坏。

6.6 在实施前对每个终端设备进行个性化设置

终端设备必须启用加密唯一身份，以确定攻击者、竞争对手和业余爱好者无法冒充生产环境中的其他用户或设备。为了充分实现这一点，必须在制造时进行个性化过程。可以通过特定 TCB 解决方案的制造商或是在印刷电路板组装 (PCB/A) 工艺中完成。

对于个性化过程，可执行以下步骤：

- 生成一个唯一密钥
- 用组织终端签名密钥（或其衍生密钥）对该密钥进行签名
- 将密钥存储在 TCB 的信任锚中
- 为该特定终端生成（或使用）唯一内部标识符
- 将唯一标识符存储在 TCB 的信任锚中
- 将唯一标识符、密钥和签名存储在物联网服务后端验证系统中

注意，终端平台个性化与网络身份个性化是分开的。由于诸多原因，采用 UICC 进行网络验证较佳，尽可能将 UICC 用作信任锚。但是，如果网络信任锚只能用于网络验证，就必须另外对应用程序信任锚进行个性化。应用程序信任锚需要具有加密唯一性，以确保在执行终端应用程序之前对应用程序平台进行验证。

如果与网络运营商或其他发行方签订了适当协议，UICC 有时可在交付之前进行置备，以用作应用程序中心信任锚。在不远的将来，终端开发人员应当评估 eUICC 技术是否适合用于物联网产品和服务。通过这些技术，可以通过类似应用程序中心信任锚的方式对加密信息进行现场置备。移动行业是个性化和置备过程的领导者，因此使用 eUICC 作为信任锚可能具有明显优势。

此外，这些技术将加入远程置备功能和安全信道，以便在应用程序和 eUICC 信任锚之间进行安全通信。这些功能可实现现场个性化，降低每个终端个性化和置备的总体成本。

附录 B 提供了有关在物联网服务生态系统中使用 UICC 卡的简短教程。

挑战来自于管理终端身份和签名过程。必须在一个无法篡改的系统中，对每个身份以及与该身份匹配的唯一标识符进行编目。虽然该过程通常是在印刷电路板组装工厂进行，但必须在工厂和企业之间建立连接，以便安全传输身份数据。

对于比较熟悉加密个性化的工厂，推出该解决方案可能十分简单。其他制造厂可能没有相应的实现流程。移动行业之所以能够以这种方式取得成功，在于其能够控制 UICC 等嵌入式技术的制造和实现。移动行业在这方面一直处于领导地位，而物联网应用程序终端的个性化和置备过程仍处于起步阶段。

请确定终端身份是否应当（或能够）由网关或上行链路进行管理。评估物联网产品或服务的架构有助于确定身份管理的这一属性是否会影响个性化过程。虽然可以向网关分配信任，但组织应确定是否能在不降低通信和验证系统整体安全性的情况下充分委派信任。

个性化相关费用通常包括但不限于以下内容：

- 芯片制造商实施个性化流程
- 制造商和物联网服务供应商协调或交付唯一个性化数值
- 实施和管理个性化身份

6.6.1 风险

如果组织选择不实施终端设备个性化，有可能会面临无法区分不同终端的风险。如果终端系统的所有密钥一致，那么序列号是否唯一并不重要。这是因为攻击者只要从任一终端析取了密钥，就可以冒充任何终端。

个性化则迫使攻击者从每个要复制或冒充的终端析取密钥，以此作为对抗方式。因为这一流程的费用可能非常高昂，所以使用信任锚进行个性化是对抗复制和冒充最有效的手段。

6.7 最小可行执行平台（应用程序回滚）

最小可行执行平台 (MVeP) 是为了创建可靠执行环境以与信任锚进行通信而必须执行的最小工作量。通常这意味着：

- 配置内部时钟或振荡器
- 配置核心外设（内存、存储器）
- 启用各种硬件桥接或外围设备
- 验证 CPU 要执行的下一个代码块
- 执行下一阶段的代码
- 管理终端应用程序映像回滚

定义 MVeP 之后，最小引导加载程序就可以使用信任锚验证更强大的引导加载程序，或者在验证外部应用程序之后执行其余引导加载程序。这样可以通过最小工作量定义一致环境，从而验证定义应用程序平台的后续代码链。

使用 MVeP 模型的另一个优点是，即使处理器只具有极低内部 NVRAM 或 EEPROM，也可以使用内部或外部信任锚引导可信架构。

最后，若要回滚至特定平台的稳定版本，MVeP 也很重要。如果可以定义一个具有验证应用程序固件映像完整性和配置执行环境所需最少功能的 MVeP，则其功能可以与核心应用程序功能分离。这样，如果任何原因导致固件更新失败，MVeP 仍可用于重新连接至后端网络并下载另一个固件映像（可以是同一个映像，也可以是较早的映像）。这也让 NVRAM 芯片损坏的终端仍能与后端服务通信并提交诊断信息。

6.7.1 风险

虽然看起来似乎没有风险，但定义 MVeP 可以确保整个终端的架构对引导过程中的每个步骤进行密码验证。为了确保终端能够向网络及其对等验证自身身份，这一步至关重要。如果 MVeP 架构设计不当，可能导致引导过程中出现安全漏洞，攻击者可能利用这些漏洞破坏安全架构。

6.8 对每个终端进行独一无二的配置

个性化保证每台设备制造出来即是唯一的，而置备则确保激活、更新唯一设备并与特定客户身份关联。置备过程有助于区分制造的设备与购买并/或部署在物联网环境中的设备。置备可以帮助物联网服务供应商：

- 区分正在使用和停用的设备
- 关联终端与网络或其他特定客户相关资源
- 根据客户需求定制终端
- 更方便地判断特定客户或终端是否已被攻陷

置备过程并不是在制造期间进行，而是取决于制造期间部署的个性化过程。置备通常在现场由启动激活过程的客户进行。但是，为了确保安全，置备需要依赖个性化过程期间设置的唯一安全令牌来确保将唯一终端绑定至唯一客户。这样，攻击者就无法仅通过猜测终端详细信息来随意注册（或注销）终端设备。相反，他们需要在个性化过程期间生成和设置的每个唯一加密令牌，这在计算上是不可行的。

通过这种方式，物联网服务供应商可以从数学上保证攻击者无法随意欺骗或注册终端设备。这样可以提高物联网环境的安全性和稳定性，让客户和设备之间的关系更加可信。

6.8.1 风险

如果不实施置备过程，可能导致组织与其终端节点失去同步。组织会更加难以跟踪终端并确定哪些设备已在生态系统中启用或停用。此外，可能也难以确定哪些设备与特定客户关联，以致更加难以在现场追踪有问题或可能被攻陷的设备。

6.9 终端密码管理

包含用户界面的设备必须能够有效地管理密码。这需要

- 缓解暴力攻击
- 禁用默认或硬编码密码
- 实施密码最佳实践
- 不允许在登录界面上显示用户凭据
- 对无效密码尝试实施阈值和增量延迟

用户需要防御最简单的攻击：被其他用户猜测密码。这只需杜绝暴力攻击可能性即可缓解。方式可以是增加密码尝试之间的时间限制。对于每次失败的登录尝试，在允许输入下一个密码之

前应设置一个更长的延迟。应当设置一个上限，限定同一时间尝试次数不能超过 N 次。否则就应实施合理的锁定时间。输入真实凭据后，应将暴力尝试告知用户。

*切勿*在物联网系统中使用硬编码或默认密码。*切勿*设置进入系统的管理“后门密码”。*切勿*设置具有默认凭据的特权帐户。为了防止一些在互联网上漫游寻找弱安全性的用户对用户设备进行未经授权的入侵，遵照以上建议是十分必要的。

密码必须满足当前信息安全最佳实践的最低质量要求。这样可以确保暴力破解密码的难度，帮助用户防范盗窃行为。可参考 OWASP 或 SANS 密码安全原则，以确保应用程序符合近期最佳实践。

切勿在用户屏幕上显示密码。务必使用星号或其他字符隐藏密码。

此外，所有接受密码的接口都必须使用暴力缓解技术。用于*验证*密码的技术也必须采取强制措施。例如，网络浏览器呈现的网页中嵌入的 JavaScript *不应*实施暴力缓解。任何精通 Web 的攻击者都可以通过互联网与后端验证服务器进行交互以绕过这些控制措施。在这种模式下，缓解技术必须在服务器端实施。在移动应用程序中，如果应用程序的安全存储区域中嵌入了本地 PIN 或密码，则移动设备必须缓解该接口的暴力攻击。

此外，在每次无效密码尝试后，缓解系统应当延长允许尝试的延迟时间。还必须为无效密码尝试设置最大阈值。达到这个阈值之后，应拒绝用户访问，等待双因素身份验证或其他更具侵入性的模式。难度

这个过程极易实施，工程团队需要承担的工作量极低。

6.9.1 风险

不实施本建议的风险包括：

- 被盗设备可能遭到暴力猜测密码破解
- “偷渡式”互联网攻击只需使用硬编码密码即可破坏物联网系统安全
- 如果用户界面显示输入到系统中的真实密码，用户会因“肩窥”受到攻击

6.10 使用经过验证的随机数发生器

请确定您的 TCB 是否能够具备真随机数生成功能。这十分重要，因为如果没有该功能，密码验证过程可能受损，使得加密数据更易猜测，削弱数据完整性。

这对于唯一密钥生成也极其重要。在一组给定的环境条件下，决不能让攻击者有能力影响环境，导致 TCB 生成在密钥、签名或填充加密信息时生成可预测数字。

此过程非常简单，只需确定 TCB 是否能够具备通过 FIPS [10]、EMVCo [11] 或通用标准认证的随机数生成功能。

6.10.1 风险

由于诸多原因，在没有强大的随机数生成器的情况下使用密码是非常危险的。原因过多，本文无法一一列举，以下列出一些需要注意的关键弱点：

- 密钥生成可能受到影响，从而生成较弱或可预测的密钥
- 一次性密码/填充或密钥可能较弱或可预测
- 用于阻挡消息重放的消息填充可能受到影响

这些问题可能导致整个物联网生态系统密码安全的整体完整性遭到大幅削弱。此风险不仅会影响终端设备，还会影响整个网络。

6.11 对应用程序镜像进行加密签名

存储在 CPU 核心 EEPROM 之外的所有应用程序都必须经过密码验证。只需按照以下步骤即可实现：

- 确定代表应用程序映像版本的元数据
- 生成应用程序映像的密码散列，包括元数据
- 验证应用程序元数据与内部元数据匹配
- 验证散列值与信任锚内部值匹配
- 用应用程序签名密钥对签名进行密码验证
- 通过密码验证应用程序签名密钥经过组织根签名

此过程的顺序是先执行最不稳定的操作，最后执行不容易失败的操作。这样，只需最低工作量即可检测最有可能的风险。

此过程极易实施，尤其是当 TCB 能够代表应用程序执行过程的开头部分时。真正的挑战更加微妙：*由哪个应用程序*执行操作。

没有经过密码验证的应用程序无法执行操作，因为它无法知晓其自身代码是否已经被攻击者破坏。改写 NVRAM 中的代码是攻击者操纵嵌入式系统的常用方法，前提是嵌入式系统不对应用程序进行验证。

内部 EEPROM 应用程序必须先对外部持久存储器中的所有应用程序映像执行该步骤。然后，该程序可以亲自执行操作，或是请求编入内部 EEPROM 中的应用程序代表它执行此类测试。

6.11.1 风险

如果终端固件 (NVRAM) 中存储的应用程序映像未经过加密签名，系统将无法区分经过授权的代码和攻击者注入的代码。这不仅可能导致攻击者得以利用可执行代码操纵以物理方式攻陷的终端，还可能使竞争对手企业能够在终端上安装自己的软件。

6.12 远程终端管理

并非所有终端都需要远程管理，但对于需要远程管理的终端，其架构设计必须能够防止第三方利用管理凭据攻陷部分（或全部）现场终端。合适的解决方案具体取决于终端的功能，但应遵循以下指导原则

- 不要将私有加密组件放在终端上不安全的存储器中，如 SSH 私有密钥、TLS 私有密钥或密码

- 尽可能为每个终端生成管理令牌（密钥或密码）
- 如果使用密码，应确保密码的使用符合密码复杂性和长度的相关最佳实践
- 尽可能为管理员实施双因素身份验证
- 管理员远程访问终端时，确保最终用户知晓
- 考虑限制虚拟专用网络 (VPN) 的管理权限
- 不要将远程管理功能嵌入可公开访问的应用程序或 API，使用分开和独特的通信信道
- 确保管理通信信道的保密性和完整性
- 通过使用行业标准通信协议，确保通信协议具有足够的熵，从而降低管理指令重放的可能性

6.12.1 风险

如未定义、实施和执行远程管理策略，可能导致终端遭到远程攻击。如果超级用户访问终端设备的权限没有严格的安全模型，攻击者可能可以对技术进行逆向工程，或从终端析取安全密钥，从而获得生态系统中所有终端的访问权限。管理访问往往是嵌入式系统中最先遭到攻击者利用的技术之一，因为它们经常配置不当或是技术薄弱。

6.13 日志和诊断

为了评估终端设备遇到的问题，物联网服务供应商应当持续评估终端的行为，确定终端的工作方式是否不超出经过认证的一组行为范围。为此，应采取以下三种策略

- 异常检测
- 终端日志
- 终端诊断

终端应记录自身行为，并定期上传日志以供后端服务处理。此类日志应由常规活动构成，如内核日志、应用程序日志和其他元数据。

此外还应定期检视诊断信息并与常规日志一同或分开传送至后端服务。诊断消息应包括尽可能多的终端环境数据，如温度、电池寿命、内存使用情况、执行时间、进程列表（如适用）等等。这些信息有助于确定问题或异常事件的发生时间以及与哪些服务相关。

网络异常检测有助于发现无法通过日志或诊断分析找出的问题，也有助于对日志或诊断中观察到的问题进行分类，或是将问题归于物理环境中出现不良反应的特定组件，例如不断重连网络的蜂窝模块或是产生不良数据的传感器。

总之，这些信息不仅有助于确定现场是否发现技术缺陷，还有利于确定异常行为是否表示有安全事件发生。

6.13.1 风险

如未实施日志记录和诊断，可能导致组织错过重要信息。这些信息不仅会影响生态系统的安全性，还可能有助于诊断重要的产品工程设计缺陷。

6.14 执行存储器保护

嵌入式系统设计通常会采用技术不够强大的微控制器：如内存管理单元 (MMU) 和内存保护单元 (MPU)。但是，任何需要实现以下功能的平台都*必须*使用这些技术：

- 运行非特权应用程序
- 运行不受信任（第三方）应用或应用程序
- 在非特权过程中运行仿真程序或虚拟机

任何需要运行非特权应用程序的环境都必须能够防御流氓应用程序或被攻陷的应用程序。这可以确保流氓应用程序或被攻陷的应用程序无法访问内存中控制特权资源的区域，如 TCB、信任锚驱动程序或硬件外设寄存器。

这方面的挑战往往是从八位微控制器平台迁移至更强大的平台，如 32 位微控制器或完整的处理器架构。但是，有很多免费或只收取极低许可费用的操作系统可供通过 MPU 或 MMU 正确实施内存保护的嵌入式系统使用。

6.14.1 风险

如果没有使用这些技术，就无法限制流氓应用程序或被攻陷的应用程序改写核心资源，如驱动程序、外设寄存器甚至是内核和其他应用程序等特权服务。缺乏内存保护会导致任何应用程序都可以完全访问微控制器或处理器上的全部内存。*必须*限制非特权应用程序滥用这些资源。

6.15 在内部 EEPROM 外引导加载

大多数引导加载程序代码嵌入在 CPU 内部的电子可擦除只读存储器 (EEPROM) 中，但也不尽然。请确定您的 CPU 是从外部源加载引导程序。如果 CPU 没有 EEPROM 验证引导加载程序代码，本地攻击者可能会对其进行操纵，以利于攻击者的方式配置 CPU。

根据承载引导加载程序的芯片或内存区域受到的保护级别，攻击者可能使用本地总线处理器（如串行外设接口 (SPI)）或远程 API（如远程固件更新）操纵嵌入代码。如此一来，攻击者就能够在最受信任的执行点即可执行代码的第一阶段放置自定义代码，从而攻陷计算平台。攻击者还可以采取另一种攻击手段，通过拆焊再焊上新的芯片，直接将引导加载程序芯片换成含有定制指令的芯片。如果没有办法验证外部代码的完整性，用户就无法区分经过认证和未经认证的软件。

为了定制引导加载程序，攻击者需要开发或外包开发引导加载程序。这一操作的难度范围可以从极其容易到极其困难，具体取决于可用资源和目标处理器。

可考虑使用带有内部 EEPROM 或可锁 NVRAM 的 CPU 或 MCU/MPU 来存储引导加载程序。这有助于确保平台至少能够验证架构加载和执行的第一个可执行内容，从而提高设备的可信度。

6.15.1 风险

如未评估信任链并确保 CPU 加载的初始代码完整性得到验证，可能导致整个系统遭到攻击。这个步骤对于保护物联网终端设备和生态系统而言十分关键。

6.16 锁定存储器的关键部分

内存可执行区域存储的关键应用程序，如第一阶段引导加载程序或可信计算基，应当以只读方式存储。这可以确保将设备引导至未被攻击者注入的有效配置。如果没有这一保障，执行第一阶段后加载的可执行代码将无法确信自身已被引导至有效配置或状态。

虽然事实上攻击者仍能通过用自己的代码替换这些关键内存区域来破坏系统，但需要他们自己开发定制版本的软件，这个过程可能非常复杂和困难。这大大提高了攻击的总体成本和成功所需的技能水平。此外，如果使用了个性化和置备，这些步骤会迫使攻击者对每个终端都重新进行这一过程，根据本地系统的唯一密码特征定制他们的解决方案。这会大幅提升攻击总体成本，降低可行性。

若要修复这一风险，只需确定关键部分内存的存储技术能够锁定。也可采用可锁定的 EEPROM 技术。

如果使用锁，确保不是在软件中设置。软件定义的锁只会在软件执行了接合锁的相应功能之后才会启用。这会存在一个几毫秒的时间窗口，攻击者可以在此期间利用解锁状态实现其目的。因此，应当尽可能采用保险丝或比特锁等硬件锁。

6.16.1 风险

攻击者可以轻松改写未处于锁定或只读状态的内存关键部分。他们可能因此获得无需进一步操作即可攻陷整个终端平台的权限，破坏系统使用的所有后续安全控制措施。

6.17 不安全的引导加载程序

引导加载程序的任务不仅是配置 CPU 执行主应用程序，还有加载执行控制并传送至应用程序。为此，引导加载程序通常会寻找主应用程序并将其加载至 CPU 主存储器中。在某些类型的系统中使用默认引导加载程序时会出现问题。

例如，微控制器供应商使用的许多引导加载程序允许将外部固件加载至 CPU 存储器以供执行，或者允许固件通过串行接口更新。其他引导加载程序可能会提示用户包含应用程序映像的位置，允许用户执行他们选择的任何应用程序。

虽然台式电脑、笔记本电脑甚至服务器等环境中经常使用这一功能，但对于嵌入式系统是不能接受的。这是因为，如果引导加载程序加载并执行了未经验证和不受信任的应用程序，则所执行应用程序的可靠性和安全性得不到保证，导致嵌入式设备处于可疑状态。

因此，为了修复这一问题：

- 引导加载程序必须能够对要执行的应用程序映像进行密码验证
- *不*应使用允许替代映像或刷固件的默认/标准引导加载程序
- 引导加载程序不得允许从任意存储位置加载应用程序映像

- 第一阶段引导加载程序的可执行映像应锁定在 EEPROM 中，只能通过安全过程进行更新

此外，引导加载程序的设计应当由第三方安全分析师进行审查。如果有人通过操纵软件漏洞攻陷引导加载程序，可能可以执行定制代码或绕过完整性验证检查。这可能引发对企业不利的*越狱行为*。确保对系统使用的所有引导加载程序进行彻底审查，排除可能导致安全风险的软件编程漏洞。

6.17.1 风险

不安全的引导加载程序的危害程度与架构设计不佳的引导加载流程一样。保护引导加载程序是确保物联网终端完整性的一个关键步骤。

6.18 完好的正向加密

完美前向保密（PFS）针对是在两个终端之间建立通信期间所交换密钥的披露。终端通常使用非对称证书验证其身份。验证阶段结束时，会使用非对称加密保护密钥协商，以生成并商定对称密钥。密钥生成并商定之后，将用于保护两个实体之间的其余会话，以此降低非对称加密产生的计算费用。对称加密的计算费用较为低廉，这意味着用于嵌入式或低功耗技术时更加快速和省电。

不过存在一个问题。这种常用密钥协商模型的前提是非对称密钥*始终保密*。然而可能并非如此。将来，资金充足的实体可能能够为任何给定的公开非对称密钥计算私有密钥。如果攻击者*保存了目标实体及其对等体之间的所有通信会话*，该实体将来可通过生成私有密钥来解密*过去的所有通信消息*。

此外，服务器密钥可能受到匿名第三方甚至企业内部人员的攻击。如果发生这种情况，如果任何人曾经存储过由被盗非对称密钥保护的通信消息，现在都可以解密这些消息。

此问题的一个解决方案是在密钥协商过程中生成一对临时非对称密钥。只有这对临时密钥的公开密钥会传送至通信链路的每一侧，用于传输对称密钥。

该临时密钥生成时的熵值和大小应足以在一段合理的时间内阻止穷举计算攻击。这样可以确保密钥协商过程可持续，未来不容易受到攻击。

此外，这种方法可以确保对等体只将持久非对称密钥用于身份验证，不会用于保密性和完整性。如果非对称密钥遭到窃取或泄漏给公众，只会影响身份验证过程，不会影响到通信信道的保密性和完整性。

为使这一过程抵抗攻击的能力更强，用于身份验证的非对称密钥必须经过安全吊销处理，以确保终端能够验证密钥是否曾经泄露。如果终端收到通知曾经发生泄露，就不应再在身份验证时信任该密钥。

6.18.1 风险

如果攻击者曾经获得权限访问保护通信信道的私有密钥，不实施 PFS 可能会导致该攻击者接触到所有网络通信。如果攻击者在未来的任何时候截获了私有密钥，都可以对其过去截获的所有通信进行解密。这会导致严重的后果。

6.19 终端通信安全

虽然本指南其他几项建议和风险中已经有所涵盖，仍然要简明扼要地指出：终端通信安全是物联网终端面临的最大威胁。操纵通信信道是攻击者攻陷终端最简单的方式。

因此，终端设计人员必须从以下角度实施通信安全：

- 验证网络对等体身份
- 数据保密性
- 消息完整性

虽然为了与其他组织设计的终端进行交互操作，可以发送和接收明文信息，但包含指令、用户隐私数据或关键系统消息的数据在通过*任何*信道传送时都*必须*得到保护。首先需要验证对等设备的身份，确保与其声称的身份一致。这一点在对等体代表系统服务时尤为重要。

其次，需要对数据保密以确保第三方无法读取通信信道传送的关键数据。

最后，需要保持消息完整性，以确保机密消息没有受到攻击者篡改。

这三个属性结合在一起，可以让通信模型在几乎无需工程变更的情况下持续使用数年。

使用经充分分析的现有安全协议还能进一步简化此过程，这些安全协议包括但不限于：

- 最新批准的 TLS 标准
- 最新批准的 DTLS 标准
- SSH2 用于身份验证和密钥交换的
- GBA 用于密钥生成和交换
- OAuth2 用于授权
- BEST, Battery Efficient Security for very low Throughput Machine Type Communication (MTC) devices [21]

虽然工程团队可以使用任何符合上述要求的协议族，但采用标准通信协议族可以降低现场发生的错误数量。这是因为信息安全和密码学领域的专家也会参与标准化协议的制定。

有关基于 3GPP 的蜂窝通信技术（包括标准化 LPWA 网络技术 NB-IoT 和 LTE-M）安全属性的详细信息，请参阅 GSMA PRD CLP.14 [4]。

6.19.1 风险

毋庸置疑，通信安全是必需的，但它*之所以*必需的原因有时会令人困惑。通信安全不仅能保证攻击者无法读取数据。它还能保证：

- 无法冒充终端
- 无法冒充关键服务
- 可以检测到滥用消息
- 软件或安全配置的变更可以安全进行

如果无法保证通信安全，就无法保证物联网产品或服务的质量、可靠性和隐私。

6.20 验证终端身份

如果每个终端都拥有加密的唯一身份，如唯一序列号，该设备必须能够证明它*确实代表该序列号*。为此，TCB 必须使用仅 TCB 和物联网后端服务知晓的密钥对消息进行加密签名，其复杂程度可由 GBA 等技术控制。该消息应包含终端对应的唯一身份（序列号或其他令牌）和元数据。

由 TCB 签名的消息必须也包含后端系统发出的质询。这可以阻止攻击者*重放*已经从 TCB 提交至后端的身份验证信息。如果质询中包含足够的*熵*，就可以杜绝消息重放的可能性。

质询终端身份的步骤如下：

- 从终端接收一个包含唯一身份令牌的请求
- 生成一条唯一质询并将其发送至终端
- 从终端接收包含签名和消息的质询应答
- 使用共享密钥验证签名正确
- 确保签名信息包含正确的身份令牌以及其他任何相关元数据
- 确认已验证的签名

处理质询的步骤如下：

- 连接至后端系统
- 接收后端系统的加密身份
- 使用 TCB 对后端系统的身份进行密码验证
- 将一条包含终端身份和其他元数据的消息发送至后端
- 从后端接收一条质询
- 生成一条包含唯一身份令牌、元数据和质询的消息
- 对消息进行签名
- 将消息及其签名发送至后端
- 确认后端系统批准签名消息

6.20.1 风险

不实施本建议的风险是终端可能会被复制或是容易受到冒充攻击。这会使组织的基础设施遭受竞争对手和攻击者的攻击。竞争对手可以利用终端身份验证的缺乏，按照同一份材料清单以更低的成本建立竞争平台。

竞争对手还可以利用验证的缺乏出售外挂在组织基础设施上的硬件。这些问题会导致企业收入的损失和运营成本的上升，因为竞争对手不用付费就可以利用企业的网络基础设施获利。由于网络带宽的成本可以量化，而云服务器、CPU 使用、磁盘使用和其他资源的成本都可以量化，这种寄生企业可能会对弱势组织造成严重影响。

7 高优先级建议

高优先级建议是指只有终端架构需要时才执行的一系列建议。例如，并非所有终端架构都要求防篡改产品保护。应当对这些建议进行评估，以确定业务案例是否需要。

7.1 使用内部存储器处理加密信息

处理器应当尽可能使用内部 CPU 存储器处理核心加密信息和信任锚未包含的密钥。这可以确保监控或有能力操纵内存总线的攻击者无法获取核心加密信息，只能看到正在运行的应用程序使用这些加密信息的效果。

此模型可以延长加密信息的寿命，迫使攻击者放弃窃取这些加密信息。作为替代手段，攻击者需要操纵 RAM 中与使用这些加密信息*效果*等同的位。如此以来，每次内部使用加密信息时，攻击者都需要更改存储器中的位，大幅提高了攻击的复杂程度。

并非所有操作系统都定义了利用内部 RAM 处理加密信息的模型。因此，工程团队可能需要亲自实施。这一过程虽然不难，但也不可掉以轻心。可执行代码必须确保其存储例程全部使用确保代表内部处理器内存的特定区域。这可能需要额外的工作量，具体取决于操作系统和使用的编译工具链。

7.1.1 风险

多数微处理器和一些 CPU 都有一小部分内部 SRAM 专门用于从内部 EEPROM 或内部 NVRAM 运行代码。外围设备一般无法访问这种 SRAM，除非是有意利用 DMA 等技术接触。如果将其保密，有能力拦截 RAM 通信的攻击者接触到代码处理的加密信息的可能性就会大大减小。

虽然风险不高，但加密信息仍不应通过公开访问总线传递，这样可以降低受到攻击的可能性。设备齐全，有能力拦截高速 RAM 通信的攻击者*能够*截获加密信息等数据，但需要熟练的逆向工程师才能截获 RAM 中与加密操作有关的消息。

因此，虽然这条建议很重要，但对于确保物理安全而言可能不是十分关键。如果将核心密钥存储在信任锚中，且应用程序仅处理会话密钥，那么在外部 RAM 中处理密钥就不太可能导致立即泄露。但其前提是密码架构只会暴露对于核心物联网操作重要性较低的密钥，如密钥轮换、会话密钥生成和证书吊销。

7.2 异常检测

终端行为建模是物联网安全的一个必要组成部分。这是因为，如果只记录和分析与设备之间成功的交互，正常运行的终端可能无法辨识被攻陷的终端。从更综合的物联网环境角度来看，应对设备的全部行为特征进行编目，以识别可能反映攻击行为的异常情况。

终端的异常行为可能包括以下行为：

- 不规则重启或设备重置
- 反复不定地离开或加入通信网络
- 连接异常服务终端，或在不正常的时间连接服务终端
- 网络流量特征明显异于正常情况
- 终端向服务器终端发送多条格式错误的消息

如果物联网服务供应商对终端类型的正常行为进行了编目，组织就能够识别可能反映异常行为的行为模式。通过设置行为基准，然后连续监视潜在异常值，组织可以更快地诊断生产环境中的安全和性能问题。

对行为特征进行编目也可能有助于组织快速将一系列功能故障与特定功能或环境条件关联起来。相对没有收集行为数据的情况，这样可以更快地制定工程解决方案。

7.2.1 风险

如果没有异常检测，可能需要很长时间才能检测到物联网生态系统中被攻陷的终端。如果只能在常规操作之外发现终端的异常行为，管理团队可能找不到不信任终端的理由。但是，如果对整个生态系统实施异常检测，可能可以及早检测到恶意行为并进行遏制。

7.3 使用防篡改产品外壳

物理设备不仅需要具备芯片级的防篡改能力，还需要产品级的防篡改能力。产品使用的外壳应能防御有攻击性的用户或是好奇用户。实现方法有以下几种：

- 可在开启外壳时使 NVRAM 无效的电路
- 可在检测到光线时熔断安全保险丝的传感器
- 可在静态设备位置移动时触发警报的传感器
- 用环氧树脂覆盖核心电路元件
- 从设备拆除内部或可拆卸组件时发出警报

使用这些方法可以提高物理终端的防篡改能力。但是，改进电路本身的设计可能更具成本效益。虽然这些方法可以成功降低业务爱好者或攻击者攻陷设备的可能性，但是无法抵御设备齐全、经验丰富的安全分析人员。

因此，这些方法可以帮助组织确保产品本身不在拥有它的消费者手中时不会受到篡改。换句话说，如果消费者把设备放在家里或现场，攻击者不仅要设法实际接触设备以实行攻击，还必须攻陷防篡改安全措施，才能修改并取代设备。这样可以防止设备遭到快速攻陷和取代，为物理设备的安全性带来有价值的提升。

但是，如果威胁模型忽略了这个方面，专注于修复来自高级攻击者和设备齐全的攻击者的一般物理攻击，就无法完全修复这种威胁。在这种情况下，这些防篡改附加措施可以拖慢攻击者的速度，但是无法阻止有充分时间和专业知识的攻击者。

因此，必须在成本效益和特定设备的威胁模型之间取得平衡。自动取款机 (ATM) 就是此类设备的一个很好的例子。外壳防篡改能力是保证 ATM 安全所必需的，它可以确保攻击者无法打开和修改物理外壳，从而截获词条数据并记录接入号码。但是，精明的攻击者已经开发出本地类似组件“分离器”，在现有 ATM 之上进行调整。因此，物理防篡改只能达到部分效果。应用程序和硬件设计必须采取其他步骤来抵御物理攻击。

工程师和企业管理者应当对给定产品或服务的威胁模型进行评估，在攻击风险和设备实施的防篡改措施之间取得平衡。每种类型的防篡改能力都会产生成本，具体取决于涉及到的过程、工程设计和材料。但是，投入并不一定就能达到期望的安全水平。

例如用环氧树脂涂覆芯片。虽然这种处理耗资不菲，但攻击者有两种方法可以轻易绕开环氧树脂：

- 从环氧树脂覆盖的元件分接电路
- 去除环氧树脂

虽然环氧树脂掩盖了芯片元件，但它没有也不能阻止电子在树脂涂覆芯片接出的电路中移动。因此，如果关键加密信息是通过硬件总线传送，环氧树脂将无法阻止攻击者拦截这些数据。

此外，环氧树脂本身也很容易去除。过去几年间出现的一些爱好者自主研发技术清楚说明了一种实用方法，使用消费者易得的化学药品和过程即可去除电路上的环氧树脂。这一过程可能具有腐蚀性和危险性，但由熟练的反向工程师描述的步骤非常完善，任何人只要拥有一间通风良好的实验室或办公室就可以实施。

因此，必须进行风险评估，以明确权衡防篡改技术的优势与攻击的难易程度。如果每台设备只需防范希望轻松操纵和随机入侵设备的攻击者，就应当采用防篡改措施。如果必须要阻止高级攻击者拦截硬件总线上的消息，那么就应当考虑为应用程序和操作系统采用比防篡改更有弹性的安全架构。

7.3.1 风险

正如前一节所述，根据对设备的要求不同，不部署防篡改措施的风险会有很大变化。如果需要设备在物理设备遭到打开、破坏或修改时提醒用户，那么防篡改措施就很重要。如果需要防止业余或熟练的安全研究人员或攻击者入侵设备，架构安全可能是该风险的最佳解决方案。

无论在哪种情况下，不部署防篡改措施都会导致用户无法确定攻击者是否曾篡改过物理设备。对于具备稳定和强化硬件及应用程序安全架构的应用程序，这可能不会有很大的影响，但对于为用户提供关键服务的产品，如医疗设备、远程信息处理系统以及家庭安全或自动化系统而言，这会产生*很大*的影响。

7.4 确保信任锚发送和接收通信的保密性和完整性

信任锚的所有通信都必须经过身份验证，并确保机密性和完整性。这个模型唯一的例外是信任锚位于处理器核心内部时。任何外部信任锚，例如 UICC，只有在接收和发送的消息可信任时方可信任。

为此，应选择能够验证身份和加密的信任锚，并确认所有包含质询应答的消息发送时均予以保密并尽可能具备可验证的完整性。

可通过安全信道管理的 UICC 具备保密性和完整性功能。物联网服务供应商应与网络运营商商讨是否能够采用 UICC 安全信道技术提高应用程序安全性。将来 eUICC 会具备应用程序安全功能。到那时，就可以使用安全信道来确保从引导加载程序阶段到网络身份验证阶段的终端应用程序安全。

虽然这个过程看似简单，但也有其微妙之处。必须对通信层的各个方面进行测试。来自各个信任锚的一些信息可能没有保密或具备完整性。例如，一条反映操作成功或失败的消息看起来似乎无害，但必须得到保护，以确保攻击者不会发送定制应答以欺骗应用程序。

一些信任锚可能无法保证通信信道中的完整性。作为最佳选择，应当采取完整性措施以确保消息未经篡改。但是，这要求主机处理器和信任锚都具备信任基础，对于应用程序而言可能是不合理的。

由于设备齐全的攻击者攻陷可以攻陷任何嵌入式系统，所以仅仅为了本地总线通信就要求两个处理器都具备信任根可能有些过分。但是，对于非常需要物理安全性的应用程序，应当实施安全性措施。

7.4.1 风险

没有确保机密性和完整性的风险较为特殊。这种风险可能高到整个系统遭到攻陷，也可能低至无害的信息收集。这是因为某些消息*可以*被修改。例如，如果 TCB 要求信任锚验证一条消息的完整性，它会通过硬件总线将消息传递至信任锚。

如果信任锚在 CPU 内部，攻击者在没有先进昂贵设备的情况下很难修改消息。但是，如果信任锚是电路板上的一块独立芯片，攻击者就可能有机会通过分接电路并插入自己的硬件来修改消息。如果信任锚接收消息后仅回复“是的，这条消息有效”，却没有验证完整性，TCB 就无法验证消息是否曾经遭到能够实际接触总线的访问者的篡改。

此外，即使应答*经过*完整性验证，能够实际接触总线的攻击者也可以轻易入侵电路，截取 TCB 的消息请求，将自己的受信任消息发送至信任锚，将真正的信任锚响应传送至 TCB。如果硬件通信总线防护不力，也可能遭到这种攻击，使信任锚无法执行任务。

因此，希望 CPU 和信任锚同时具备独立的内部信任锚是自相矛盾的。如果一个可引导的 CPU 可能遭到攻击者的篡改，而 CPU 又必须使用自身 EEPROM 验证信任锚的完整性，那么它如何信任自身呢？这是一个难题，但是可以解决。

一种解决方案是在 CPU 的 ROM 中插入一个公开密钥。此密钥可用于验证信任锚所发送消息的完整性。如果（需要验证的）任意消息通过硬件总线传送至信任锚，信任锚可以用一条*包含原*

*始消息*的签名消息作为应答的一部分进行应答。这可以验证消息确实来自信任锚，且正在处理的消息确实是要处理的消息。唯一还需要注意的是要确保消息填充中使用的随机数能够确保加密消息不可*重放*。

综合上述内容不难发现，加密技术本身和支持加密通信的算法中非常微小的问题都可能导致加密失败。这正是（正确）实施保密性和完整性如此重要的原因所在。

7.5 无线应用程序更新

远程更新终端*应用程序*映像可以是一个简单而直接的过程。其复杂性源于解决方案的过度设计并未真正解决现实安全漏洞。从持久存储器的角度来看，设计过程十分简单：

- 为正在使用的应用程序映像定义位置
- 为备份应用程序映像（如有）定义位置
- 为紧急应用程序映像定义位置
- 如有备份应用程序映像空间，则用正在使用的映像更新此空间
- 使用 TCB 中存储的签名对正在使用的映像进行密码验证
    - 这样可以确保存储介质没有损坏，同时确保攻击者没有在写入过程中修改位
- 全部下载或增量式下载新映像及其元数据和签名
- 使用增量修补正在使用的图像
- 使用 TCB 验证密码签名
- 重新引导进入新映像

如果过程在任何环节失败，应将系统恢复至备份映像以确保应用程序根据需要执行，或者使用紧急系统进行*自动通报*，通知物联网服务生态系统发生故障。

难点在于要建立可以解决以下两个问题的存储模型：

- 企图操纵更新流程的攻击者
- 异常硬件

如果没有备份系统或紧急分区，设备除了失效之外别无选择。由于嵌入式系统通常没有强大的用户界面，可能会导致企业和客户之间出现严重的压力点。失效时提供尽可能充分的理由对于用户信心和系统可靠性而言都是必要的。

需要注意，一些攻击者可能蓄意破坏更新过程，迫使系统持续处于易受攻击的状态。例如，在应用程序的当前版本中发现了可以利用的漏洞，但应用程序的最新版本提供了补丁。

这个模型的优点在于，即使攻击者破坏了网络协商过程，后端系统仍有机会记录这个事件。如果后端网络确定某个节点*除了*更新之外的通信都正常，可能会提醒管理人员判断该终端节点是否遭到入侵。

7.5.1 风险

如果 OTA 应用程序更新流程架构设计不当，可能导致攻击者向终端远程注入可执行代码。如果攻击者在网络上占据了特权位置，可能同时影响数以千计的终端。攻击的结果可能只是简单的代码执行，也可能是拒绝服务（让终端变砖），或是彻底改变终端设备的用途。

7.6 设计不当或未执行的相互验证

在通信环境中，对等体之间通过协议的*身份*相似性进行对话。这在不同的场景中有不同的含义，但在每个环境中相同类型的*地址*标识着消息的目的地。任何实施给定协议的通信模块都可以声称自己是*特定地址*的所有者。即使设计或强制某个协议的特定*实施*使用本地无线模块的硬件地址，也没有规则规定用户可以实际改写该模块的 EEPROM 并更改硬件地址。即使该实施不允许用户动态更改硬件地址，仍然可以对其进行控制以更改地址。这一功能的结果本质上是欺骗：或冒充其他计算机身份以拦截发往该计算机的消息。

7.6.1 客户端身份验证

欺骗可能发生在所有环境中。例如，任何蜂窝无线电都可以发信号表示它是任何给定国际移动用户识别码的所有者 (IMSI)，无论真假。任何笔记本电脑都可以更改其以太网地址，冒充局域网 (LAN) 上的其他计算机。无论拓扑穿越的是物理空间还是电波空间，都可以冒充通信终端的*身份*。

身份验证可以防范这一问题。例如在蜂窝网络中，任何拥有相应设备的人员都可以随意选择声称自己是任何 IMSI 的所有者。但是，蜂窝网络运营商可以将一个独特于该用户的密钥 (IMSI) 编码到用户身份模块 (SIM) 中，以执行*身份验证*。如果蜂窝设备与基站通信时声称代表特定 IMSI，基站会发送一条只能用置备该特定身份的 SIM 卡中存储的唯一密钥解答的密码质询。如果攻击者无法解答密码质询，基站就可以确定攻击者*不*代表有疑问的 IMSI，从而阻止该用户关联网络。

上述模型描述了*基于客户端的身份验证*。在这种模型中，只要客户端能够通过密码验证其身份，几个子系统（包括基站）就允许客户端（终端）加入和离开网络。但是，有一个逆向问题可能导致客户端被操纵：*服务器身份验证*。

7.6.2 服务器身份验证

3GPP 模型只会对终端（在 3GPP 中称为用户设备）进行身份验证。终端不会对其连接的基站进行身份验证。因此，任何基站都可以声称代表任何蜂窝网络运营商。如果有人能够操纵或搭建蜂窝基站，他们就可以选择冒充任何蜂窝网络运营商。目前搭建定制蜂窝基站的成本低于 1000 美元，但产生的功率仅能拦截本地区域中的消息。建立假塔台之后，基站就可以冒充本地蜂窝网络运营商，拦截本地区域中终端的通话、文本消息甚至数据。

UMTS 和 LTE 等较新的 3GPP 网络协议实行了实体之间的相互验证。这样终端就可以通过密码验证基站是否代表其声称代表的蜂窝网络运营商。此时竞争者必须破解蜂窝网络运营商的密码才能冒充基站，大大增加了攻击的复杂性、难度和成本。

7.6.3 蜂窝读写器或假基站

然而这条规则也有例外，例如蜂窝读写器。此类设备通常由政府承包商、政府和情报服务使用，采用特定蜂窝网络运营商为这些实体提供的密钥进行编码，以保障国家安全。此类设备使用这些密钥来被动拦截双向通信，或者对特定目标主动执行中间人攻击。

但是，在现代通信威胁模型中，这种技术的使用并不局限于政府和情报领域人员。现在，这些系统只需几百美元即可搭建，从而得到能够拦截或冒充蜂窝通信的低成本假基站。

7.6.4 通信安全是门到门安全

提到蜂窝读写器有助于为本节作一个充分的总结，这让我们知道通信安全并不是绝对的，它只能保护两个实体之间的通信信道。然而这些实体扮演着门的角色，让数据可以进出这些实体所连接的生态系统。

例如，油井监控设备等工业控制系统中可能会使用特定 SIM 卡。SIM 卡设计为一个可拆卸组件。任何能够实际接触油井监控设备的人都可以取出 SIM 卡，插入笔记本电脑。如果这台笔记本电脑上有能够模拟油井设备功能的软件，后端服务器将无法区分实际油井设备和笔记本电脑。但由于有了 SIM 卡，这台笔记本电脑可以通过蜂窝网络的身份验证！这样，蜂窝无线网络验证的是 SIM 卡，而不是笔记本电脑。

7.6.5 相互验证的解决方案

物联网生态系统中的每个对等体都必须对加入该生态系统的所有其他对等体进行身份验证。为此，必须使用 TCB 来确保通信技术由正确的密码架构驱动。如果密钥很容易被攻击者获取，就无法实现相互验证。更多信息请参考本文档关于 TCB 的章节。

通过身份验证之后，每个对等体必须对发送至网络中其他对等体的消息进行加密和签名。每个接收到消息的对等体必须先对数据进行密码验证，再进行操作。并非所有通信协议都能进行相互验证或具有强密码，所以应用程序工程师应当设计足以确保保密性和完整性的协议，而不是依赖于通信协议。

即使 LTE 等更加强大的协议采用了相互验证，也无法保障蜂窝通信网络之外的基础设施安全。只有更高层级的协议安全才能解决蜂窝网络运营商控制范围之外基础架构漏洞的风险。

7.6.6 风险

未实施应用程序强安全性的风险是终端必须信任通信层的安全性。正如本条建议所述，仅仅依赖网络来解决应用程序的安全问题可能是不够的。即使 MNO 可以信任，在数据到达物联网服务供应商所有的服务器之前，消息必须途径多个并非由 MNO 所有或控制的网络基础设施。因此物联网服务供应商会面临的风险是，任何能够控制这些系统的人都可能拦截、改写或编造发送至或来自终端系统的消息。

7.7 隐私管理

物联网技术必备的一项能力是连接真实世界与数字世界。这会导致隐私问题，因为用户的真实环境与他们的喜好以及在线查看的内容直接相关。随着时间的推移，可能会带来不良影响。

因此，物联网服务供应商必须考虑其消费者的隐私，开发隐私管理界面并尽可能将其集成到终端以及产品或服务的 Web 界面。

这项技术应允许用户确定系统使用的是哪些隐私信息，了解服务条款以及停止向企业或其合作伙伴暴露这些信息。这种粒度和选择退出系统有助于确保用户有权利和能力控制其分享的与自身和现实世界相关的信息。

7.7.1 风险

不保护消费者隐私会带来很多潜在风险。跟踪、骚扰、资料分析、威胁等诸多问题都是没有保护用户数据导致的实际后果。

7.8 隐私和唯一终端身份

每个终端都有一个数字指纹。该指纹由特定终端唯一的*地址*、*序列号*和*加密身份*组成。但是，这些凭证还可以将设备与特定客户、位置或服务相关联。这在许多情况下是不利的。例如，如果智能手机在主动扫描 802.11 接入点时使用了手机的内置 Wi-Fi 地址，就可能因此受到追踪。随后当这些地址在不同位置之间传递时，就可以对其进行追踪。如此一来，任何人都能将特定 Wi-Fi 地址与特定用户相关联，并观察他们在世界各地的动向。为了解决这个问题，智能手机软件制造商在扫描接入点时生成随机的 Wi-Fi 客户端地址，这样就几乎无法通过这种方式跟踪手机。

与此相似，可以通过蓝牙低功耗 (BLE) 地址、802.15.4 地址、Wi-Fi 甚至是蜂窝 IMSI 对物联网终端进行追踪。物联网服务供应商在开发终端技术时，应尽可能使用随机无线地址连接新的环境，保证用户隐私不受侵犯。

对于 SSH 公钥等密钥也是如此。虽然用户常常希望公开其公钥，但终端上的公钥往往会暴露特定终端的用户身份，这是*不*可取的。相反，用户应当能够选择在连接到新环境时是否希望公开自己的身份。

7.8.1 风险

如果不能充分降低这一风险，移动终端用户在其设备离开和加入网络时会受到追踪。这会导致严重的隐私问题，法律团队、立法者乃至保险公司目前都对此十分关注。如果没有充分保护隐私以避免追踪行为，可能会导致物联网服务供应商在不久的将来面临法律制裁。

7.9 按照适当权限级别运行应用程序

终端上运行的应用程序通常不需要超级用户权限。应用程序经常需要访问的是设备驱动程序或网络端口。虽然此类设备、端口或其他对象中有一些在初次访问时可能需要超级用户权限，但在后续操作中不会再需要超级用户权限。因此，最好只在应用程序启动时使用超级用户权限访问这些资源。然后就应当删除超级用户权限。

删除超级用户权限是一个有据可查的常用过程，在安全外壳 (SSH)、apache2 和其他设计良好的服务器等应用程序中得到了极好的实施。这个过程通常包括以下步骤：

- 使用高权限启动应用程序

- 访问所有需要高权限的资源
- 确定应用程序运行本应使用的用户身份（例如 UNIX 用户 ID 和组 ID）
- 将过程身份完全转换为目标用户/组 ID，删除正在运行的应用程序的超级用户权限

在 *privsep* 的 SSH 实施中可以看到一个更加复杂的模型，其中一个特权服务的唯一用途就是在目标用户/组身份下引导加载主应用程序。这样，关闭服务之后即可方便地重新启动，不会对特权资源造成影响。

有关详细信息，请参考：SSH 权限分离：
http://www.citi.umich.edu/u/provos/ssh/privsep.html

7.9.1 风险

使用较高权限级别运行应用程序时，如果有一个应用程序被攻陷，可能导致整个系统被攻陷。由于超级用户权限授予了应用程序完全访问整个运行系统的权限，一旦攻击者攻陷这样的应用程序，就无法再对其进行遏制。删除权限有助于遏制攻击者，限制他们在嵌入式系统中提升权限。这可能就是系统完全被攻陷和小麻烦之间的差异。

7.10 在应用程序架构中执行职责分离

终端上运行的应用程序应当拥有对应每个独特过程的不同用户身份。这可以确保一个应用程序被攻陷之后，只有成功实施二次攻击，才能攻陷相同终端上的另一个应用程序。攻击者需要进行的这个额外步骤往往是整个入侵过程中的关键障碍，它可以增加攻击终端的成本和复杂性。

例如，允许用户检索终端状态相关信息的网络服务不得通过同一个过程操纵 TCB。就服务用途而言，这项能力*超出应有范围*。这两种不同的操作应由不同应用程序处理，并在本地操作系统的不同用户 ID 下运行，这样有助于分离应用程序的职责，并在有组件遭到攻陷时降低入侵风险。

为了正确实施这一策略，必须在底层硬件架构中启用存储器保护功能，且操作系统必须具有特权级别的概念。必须限制非特权软件访问特权资源，如驱动程序、配置文件或其他对象。

服务需要发出请求才能访问特权资源，并且需要经过系统调用等受限 API，以确保所有消息格式正确且符合安全架构的要求。

多级特权的概念已经有半个世纪的历史。但是嵌入式系统并不允许用户登入控制台并运行自己的应用程序，因此它经常被忽视。这样一来，经常会将所有的服务都部署为特权用户。然而这是有漏洞的。

每个应用程序或服务都必须使用自定义的权限实施，在大多数环境中是单独的*用户身份*。通过实行不同用户身份分离职责，可以确保被攻陷的服务不会直接影响到相同系统上其他服务使用的资源。为了攻陷其他服务和用户，必须对本地操作系统实施二次攻击以提升权限。

这要求适当的规划和正确利用权限分离的良好应用架构。

7.10.1 风险

如果没有实行职责分离，终端上任何一个服务遭到攻陷后都会导致整个设备被攻陷，因为设备上运行的每个服务或应用程序都使用同一个用户和/或组身份。如果*实施*了本建议，通过网络攻陷一个低权限服务并*不会*立即导致整个系统遭到攻陷。

本建议实施起来非常简单，因此对于物联网终端的安全非常关键。需要注意的是，远程攻陷网络服务往往需要高水平的专业知识。如果需要实施内核级攻击或二次攻击来提升权限才能控制整个系统，攻击者可能没有实施攻击所需的时间、技术或设备。

这样通过简单的配置变更提升攻击难度，对于确保设备使用寿命有很大帮助。

此外，因为可以通过过程监控以及其他分析手段检测出被攻陷的服务，所以任何服务被攻陷都会提醒服务生态系统检测出被攻陷的服务。这让管理员可以在整个系统被攻陷之前采取措施保护系统，也让管理员可以在特定漏洞被大肆滥用之前诊断和修补有漏洞的软件。这样可以让企业即使在应对熟练的攻击者时也有极大优势。

7.11 执行语言安全

编程语言的安全程度各不相同，具体取决于语言的用途以及级别。一些语言的结构可以限制对原始存储器的访问，同时限制存储器的使用方式。工程团队应当确定一种能够保障应用程序运行时或最终二进制安全的语言。

应当尽可能加强编译器或运行时的安全，以防止攻击者利用漏洞。在明确定义的运行时环境中，即使是易于触发的编程漏洞也极难充分利用。其前提是已经采取增强安全措施保护应用程序执行和访问存储器的方式，并且受到系统安全增强功能的支持。

7.11.1 风险

未加强编程语言及其所编写应用程序安全的风险是应用程序容易受到攻击。一些编程体系的漏洞出名地多，例如 PHP，专业工程团队切勿使用。Python 等其他语言适合生产环境，但隐含安全风险，必须进行评估。因此，所产生的风险可能在严重到无害之间波动。工程团队*必须*应用风险评估和威胁建模过程，充分评估哪种语言最适合他们的生产环境。

7.12 实施持续渗透测试

对于大多数可能现场发布并随时配置新终端的物联网部署而言，仅在部署时执行安全审查是不够的。建议采取持续渗透测试方法，以便及早发现易受攻击的终端软件和不安全的配置。

实施持续渗透测试策略可以快速检测和提早管理所发现的威胁，加快缓解速度并缩短威胁暴露时间。

完整的持续渗透测试策略应自动定期执行以下步骤：旨在创建可访问资产清单的资产发现；资产识别和分析；已知漏洞验证和披露；不安全配置检查，以及相应的报告和有助于修复的提示。

7.12.1 风险

如果不实施持续渗透测试策略，可能只会在部署时执行一次安全审查，新终端和新配置则不曾接受评估。这种情况可能导致一系列终端容易受到攻击，暴露于风险却一直不察，最终遭到攻击者攻陷。

8 中优先级建议

中优先级建议集包括根据终端技术设计选择而定的一系列建议。例如，只有终端上运行操作系统时，提升操作系统级安全性才有效。如果终端包括单内核应用程序或配置一个嵌入式应用程序的嵌入式实时操作系统 (RTOS)，则此建议不适用。如果建议*确实*适用于终端设计，应予以实施。

8.1 实施操作系统级安全增强功能

操作系统上运行的应用程序应设计为（透明或有意地）使用底层操作系统和内核的安全增强功能，包括以下技术：

- ASLR
- 不可执行的内存（栈、堆、BSS、Rodata 等）
- 用户指针解引用保护 (UDEREF)
- 结构泄露（信息披露）保护

嵌入式系统中使用的每个操作系统都会提供这些技术的不同变体和组合，有时候名称也不同。请确定操作系统和内核能够提供的技术，并尽可能启用这些技术，以增强应用程序的安全性。

这里的挑战来自于确认各个操作系统的能力。例如，没有内存管理单元 (MMU) 的平台上运行的应用程序可能没有 ASLR 功能。但是，UDEREF 的同等技术即使是在只有一个内存保护单元 (MPU) 的环境中也*可以*实施。请评估所用技术及其功能，再确定通过架构、内核、操作系统和应用程序保护措施的组合能够达到的安全级别。

8.1.1 风险

未实施本建议会导致应用程序运行时环境更容易受到攻击。这些增强功能可以显著限制能够针对有漏洞服务展开可靠攻击的攻击者数量。

因此，如果组织开发的应用程序有一个安全漏洞，利用其可获得远程执行代码的能力，那么实施 ASLR、NX、UDEREF 等技术可以降低其遭到利用的可能性。这会限制攻击者在合理时间内展开攻击的能力，因为攻击开发人员需要使用具有挑战性的先进技术，并且需要针对每个目标进行自定义。这不仅增加了难度，也增加了实施全面攻击所需的时间和费用。

如果没有这些增强功能，使用现成的免费软件就能在几个小时之内展开全面攻击。

8.2 禁用调试和测试技术

开发产品时，通常会启用调试和测试技术，方便进行工程设计。这是非常正常的。但是，当设备可用于生产部署时，在定义批准配置之前，应从生产环境中去除这些技术。

产品部署的批准配置绝对不能包含可能被攻击者利用的调试、诊断或测试接口。这些接口包括：

- 命令行控制台接口
- 带有详细调试、诊断或错误消息的控制台
- JTAG 或 SWD 等硬件调试端口
- 用于调试、针对或测试的网络服务
- SSH 或 Telnet 等管理接口

批准配置中应禁用所有此类技术。

可由系统删除的串行接口也应当从电路板上真正拆除。但是，很多时候 UART/USART 等串行接口可以通过微控制器或处理器上的硬件引脚启用。如果这些引脚仍作为控制台启用，攻击者只需分接这些引脚就可以与控制台进行交互。拆除 DB9 接口等实际串行端口本身并不会禁用控制台。

此外，JTAG 和 SWD 等调试端口也不能仅仅通过软件禁用。应当通过改变安全保险丝或安全锁来禁用这些设备。通过软件禁用这些技术会给攻击者留出时间，让他们能够在软件禁用接口之间连接至 JTAG、SWD 或类似硬件调试接口。对于攻击者而言，这一机会窗足够他们取得成功。

8.2.1 风险

组织如未实施本建议，就相当于打开大门，允许他人析取中央处理单元中的关键加密信息。这可能会让攻击者得以将自己的固件加载到 NVRAM 或 EEPROM 中，析取或修改关键加密信息并借此进一步攻陷物联网网络或设备。

禁用调试端口是确保物联网终端完整性的一个关键步骤。但是，组织必须对禁用这些技术的风险进行评估，并结合可以对现场发现的问题进行诊断和调试的优点进行权衡。如果无法对正在运行的系统进行调试，修复产品中的生产级缺陷可能要困难得多。

8.3 基于外设的攻击导致存储器受到污染

处理系统依赖一致性来确保给定一组输入时，算法的输出可以预测。处理系统也希望组件可以可靠运行，并且写入的每一位都保持稳定，不会更改，除非被处理器更改。在封闭系统中，这一理论是适用的。如果这种模式发生异常情况，可能会损害或是直接破坏处理环境。

信息安全列出的一类异常情况系蓄意引发，以便访问在其他情况下无法访问的对象。一个可供引发利于攻击者的异常行为的方便之门是直接存储器访问 (DMA)。简单地说，DMA 是一项技

术，处理器可以使用它允许外部组件（外设）不受 CPU 干扰地访问主处理器内存。换句话说，CPU 可以允许外设直接访问内存的一个区域。该外设可以读取或写入这个内存区域。

如果处理器没有正确限制外设可用的内存区域，该外设可能能够访问超出预期功能所需的主存。换句话说，如果为外设（例如以太网控制器）分配的 DMA 区域是用作已接收以太网帧的循环缓冲区，而分配的 DMA 区域包括整个主存储器，那么以太网控制器的固件就可以任意读取和写入全部系统内存。CPU 将无法阻止以太网控制器固件写入内存。

这种攻击会带来双重结果。数据可能自主存泄露，或被编入网络数据包或应用程序信息，以便隐秘或直接外泄。此外，攻击者可以通过覆盖应用程序的可执行代码，暗中将后门（恶意软件）插入主存。

从处理器的角度来看，它几乎无法识别过于宽松的内存窗口是否遭到恶意外设的利用。为了应对这一攻击，应确认终端系统中使用的处理器是否能够将 DMA 限制为可预测的小块内存区域。如果能够限制，应确保根据每个需要访问内存区域的外设定义每个内存区域。尽可能不要对外设启用任意窗口内存。

一些处理器可能不允许对 DMA 窗口的大小或在线性或虚拟内存中的位置进行粒度限制。对于关键应用程序而言，DMA 攻击应视为物联网终端的一项现实威胁，因此应当评估是否有必要考虑功能粒度更细的另一种处理器。

如果平台公开了 IEEE1394、Thunderbolt、Express Card 或其他允许直接访问外设组件互连 (PCI) DMA 的端口，就为现成和低成本攻击打开了方便之门。

如果对平台实施基于 DMA 的攻击需要利用本地硬件组件，难度必定会增加，但对于安全犯罪行为而言，重刷外设固件以破坏 DMA 从而攻陷本地终端并非不能做到。然而，成本、时间和专业知识也是值得考虑的因素，因此这种情况下发起攻击的往往是受资助（有偿）攻击者。

8.3.1 风险

如果选择不限制外部组件利用 DMA 的能力，可能导致平台遭到完全攻陷，至少会导致终端的密钥、隐私数据或知识产权遭到析取。

8.4 用户界面安全

配备触摸屏、彩色显示器或其他接口技术等用户界面的物联网终端必须能够通过安全的方式向用户呈现信息并接收来自用户的信息。

虽然本文档已经介绍了用户界面的属性，例如密码，但还有一些比较细微的问题需要讨论。

- 报警系统
- 操作确认

如果发生异常情况，例如物理篡改或应用程序不按预期方式运行，则用户应当收到明显报警。或者，用户应当能够在用户界面中查看系统报警。

此外，由编码或一个接口到另一个接口之间的无缝过渡驱动的设备执行的所有操作都应当经过用户的确认。例如，设备摄像头读取到 QR 码，或者 NFC 或 RFID 交互请求设备连接某个 URL。在这些情况下，应当提示用户确认操作，并验证所执行的操作是可取的。应为用户提供取消操作的选项。用户应能查看给定操作的所有详细信息，包括要连接的完整 URL。

8.4.1 风险

如果没有实施本建议，用户容易受到无法检测的攻击。虽然一些系统设计人员偏好 RFID 芯片到对应产品网站的无缝过渡，但这种行为可能会产生不利影响。用户可能被迫在未经自己同意的情况下查看不良内容，或者被诱骗访问或执行会侵害其安全状态或隐私的网站或操作。

此外，如果用户难以查看报警，可能会不理解使用可能遭到篡改的设备带来的风险。这可能会危害用户的人身安全，令他们处于危险境地。

8.5 第三方代码审计

任何时候，如果一段代码（例如引导加载程序）是构建安全运行时平台的关键组件，则必须对其进行风险审查。如果引导加载程序会被攻击者控制执行不受信任的代码或绕过验证序列，那么它就毫无用处。这会令组织在技术部署中投入的资金、时间和经验全部作废，工程费用也付诸东流。

这方面的安全漏洞还可能导致竞争对手通过欺骗、API 滥用、数据拦截、设备复制甚至设备品牌再造，在与企业的竞争中占据上风。因此，代码的关键部分必须由经批准的第三方进行审查，确保技术没有遭到滥用的风险。所以，为了找到一个足以执行审查的信息安全团队，需要评估应审查哪些类型的代码。在这个模型中，这通常意味着：C、汇编语言，也可能是 C++ 或 Java。

请找到一个精通这些语言以及底层架构的团队。虽然许多信息安全团队都可以执行信息安全审查，但可以对物联网企业使用的特定平台执行审查的团队并不多。每个平台都有细微的区别，最好聘请熟悉所用平台的团队。

8.5.1 风险

虽然聘请第三方顾问评估内部开发的技术可能较为困难，但这是保障安全所必需的。这是因为，开发技术的工程师必须能够证明他们的架构。如果开发技术的工程师是唯一审查过架构的人，就无法做到这一点。工程师倾向于从他们*试图*设计和实施的架构而不是*实际实施*出发来对代码库进行可视化。因此往往需要借助第三方来发现架构和实施中可能导致安全漏洞的细微之处。

8.6 使用专用 APN

3GPP 蜂窝网络中，接入点名称 (APN) 是专门为一组已验证设备配置的专用网络。通常情况下，专用 APN（又称“安全 APN”）是仅有关联特定企业的已验证设备才能访问专用网络。企业可以通过 APN 限制允许哪些终端通过蜂窝网络连接服务基础设施。这有助于减少能够直接访问后端基础架构中的物联网服务的用户数量。

专用 APN 的另一个属性有助于阻止流氓终端滥用物联网生态系统。防火墙可以限制能够通过 APN 连接的服务或计算机。配置得当的 APN 会禁止终端相互直接连接，从而阻止被攻陷的终端通过网络基础设施转移至其他终端。

与组织合作的蜂窝网络运营商或移动虚拟网络运营商 (MVNO) 接洽以确定安全 APN 有哪些技术。可能还有其他服务可用，例如监控，将异常设备列入黑名单，将用户身份与操作绑定等。

8.6.1 风险

使用专用 APN 可以减少很多类型的攻击。例如，专用 APN 可以帮助企业减少从终端直接到互联网的连接数量。绝对不应允许终端直接连接不受信任的互联网资源。只能信任伙伴组织，应当对这些服务进行身份验证。

如果没有使用专用 APN，被攻陷的终端就可以不受限制地与任何互联网服务通信。这样，攻击者就可能利用终端对独立基础设施发起二次攻击。这可能涉及拒绝服务 (DoS) 攻击，也可能有助于向其他企业、政府或平民发起更具危险性的攻击。

然而，需要注意的是，专用 APN 并不能降低攻击者攻陷终端与专用 APN 之间通信链路的风险。此外，专用 APN 只是对后端服务起到网关的作用，并不能确保 APN 和物联网服务供应商专用网络上的后端服务之间的安全。使用 APN 可以带来很多改善，但是这些潜在的安全漏洞必须另外予以解决。

8.7 实施环境锁定阈值

嵌入式系统中的组件设计用于特定环境阈值下，包括电压水平、电流消耗、环境或工作温度和湿度。每个组件通常都有特定的额定值范围。如果设备状态低于或高于给定范围，说明该组件可能不正常，或者其行为模式对攻击者有利。

因此，必须检测这些环境变化，以确保设备应当继续运行还是关闭。不过需要注意的是，关闭设备可能是攻击者希望的结果，他们可以利用这一设计决定进行拒绝服务攻击。工程团队应当评估这一模型，以确定是关闭更好，还是尝试继续运行更好。

无论在何种情况下，该模型通常都包括

- 在电压降得太低时进行欠压和停电检测
- 电压上限电路保护，确保电压等级不超过某个阈值
- 电流限制电路，确保电流消耗不能降低或超过特定水平
- CPU、MCU 的内部温度监控以及其他监控内部水平的组件
- 还可以选择对湿度进行评估，以判断环境是否过于潮湿或过于干燥

温度极其重要，因为高温可能表示用户、环境甚至是硬件或软件问题引发了电路故障。通过监控温度，操作系统或应用程序可以关闭资源（或整个设备）以防止终端引发火灾或其他问题。

低温也可能致使设备行为发生改变。这可能会减缓电路速度，或导致其组件作出预期之外的反映。如果温度能够导致一种可预测的异常，进而对应用程序或电路产生有利于攻击者的影响，那么攻击者就可以利用这种情况。

分析温度和湿度时，锁定阈值的难度较为明显。应当通过在电路板或处理器中设置欠压和停电电路来缓和电流和电压水平。工程师可以查看芯片的电压和电流阈值的相关数值，因而很容易针对这些问题实施保护措施。

温度和湿度的操作判断则有些困难，因为攻击者无需接触实际设备就可以人为达到这些数值。就温度而言，如果达到可能表示待定安全事件的水平，设备必须能够采取充分措施降低温度。但是，在工业控制系统或医疗器械等关键环境中，设备应当尝试尽可能继续执行关键操作。只有在超过工程师和企业管理者定义的特定数值之后，设备才能关闭。

8.7.1 风险

对于电压和电流消耗，滥用的风险在于噪声干扰和其他可以从这些数值的变化中受益的边信道攻击。如果处理器实施了欠压和停电检测，就可以降低滥用的风险。否则，风险可能在于电压和电流峰值可能导致物理设备发生安全问题，或者让攻击者得以操纵噪声干扰（和类似）攻击，从而破坏组件安全。

这些问题的修复方式是在 PCB 上设置防止电压或电流峰值或降低影响组件的电路。

对于较大幅度的环境数值变化，风险在于用户安全。CPU 使用率过高或其他异常情况引起的高温可能导致烫伤、化学灼伤甚至火灾。

8.8 设置电量警告阈值

为用户提供关键服务的终端必须设置一个指示电量相关事件的报警阈值。这些事件可能包括：

- 电池电量不足
- 电池电量严重不足
- 停电事件
- 欠压事件
- 切换至备用电池事件

警告用户时必须留出足够的时间，以便他们对电量不足采取补救措施。可以设置一个表示特定功率状态的 LED，例如绿色表示正常，黄色表示不足，红色表示严重不足。

对于连接交流电源的系统，应当配置为在发生停电或欠压事件时警告用户。此外，终端还应当将这些事件记录在持久存储器中，以确保用户和管理员之后可以检索这些信息。应为这些信息附加时间戳。

这一过程的难点在于确定电池电量耗尽的速率以及通知客户电量状态变化所需的额外电量。这可以通过电气工程设计实现，对于经验丰富的工程设计公司而言应该不会太难。

8.8.1 风险

如果没有明确定义的电量警告系统，用户将无法为可能发生的重要电量事件做好准备。虽然对于计步器、定时器和其他可穿戴设备等单一设备而言，这种情况可能没有太大危害，但个人追踪器、远程信息处理系统和家庭安全系统等更加关键的设备可能会由于电量不足而受到严重影响。

8.9 没有后端连接的环境

8.9.1 方法

即使是在没有后端网络连接的环境下，终端也必须能够确保通信安全，尤其是网关或用作网关的终端。无论这种没有连接的情况是不是暂时的，网关或终端必须能够像后端系统可用时一样确保安全。

为此，所有终端必须向其传送隐私相关、配置或指令数据的对等体都必须使用 TCB 进行身份验证。TCB 可用于确保对等体发送和接收的数据来自同一组织置备的实体。这可以降低与攻击者的设备通信的可能性。

同时仍然可以通过与其他无法验证的设备通信来实现互操作。但是，向这些设备传送的信息类型应限制为互操作和非敏感数据类型。

这里的挑战来自于确定需要验证哪些终端以及与哪些终端进行明文通信。组织必须确定对哪些数据进行分类并阻止未经授权的对等体访问。实现数据分类之后，即使没有核心物联网服务的帮助，组织也可以确定哪些对等体是合理可信的。

8.9.2 风险

为无通信环境部署解决方案的风险在于竞争对手有机会滥用基础设施。竞争对手可以通过提供互操作性和使用无连接网站作为试验场所来抢走业务。

相反，组织可以选择允许互操作，但将其限制在一定程度内。这样就可以保留某些核心知识产权和服务，仅限经过 TCB 身份验证的对等体使用。这可以帮助企业减少知识产权问题和竞争对手。

8.10 设备停用和废弃

所有终端设备都有生命周期，正如本文其他部分所述。一些设备必须停用是因为用户取消了订阅，另一些设备必须停用则是由于异常或攻击行为所致。无论出于何种原因，企业必须做好准备以使用 TCB 和通信模型安全停用设备。

如本文其他部分所述，废弃是指停用整个设备网络以及支持这些设备的服务。对于企业已经弃用或是决定关闭的产品或服务，其设备和网络必须废弃，以免攻击者接管废弃网络并加以利用。

为此，必须使用 TCB 和支持协议。通常，该过程包括以下步骤：

- 在服务生态系统创建一条停用消息
- 根据接收消息的特定终端定制消息内容
- 使用停用 PSK 或非对称密钥对消息进行签名
- 将消息推送至终端
- 从终端接收一条加密确认停用的消息

- 在验证设备列表中作废终端
- 禁止该终端继续通信

设备端软件运行的应用程序应当

- 通过服务生态系统连接至关键后端服务
- 向服务查询关键消息
- 接收停用消息
- 使用 TCB 和信任锚验证消息签名
- 生成确认消息并使用个性化 PSK 或非对称密钥对其进行签名
- 执行停用操作
- 将消息发送回关键服务

在停用之前，必须对消息进行签名并准备好传输，因为停用过程中需要从信任锚中作废并删除安全密钥。这一过程之后，用于对停用信息进行签名的密钥将不可用。该服务要求接收一条可验证完整性的消息，以确保终端确已接收并处理过消息。

这一流程的难点主要在于，停用可能被攻陷的设备需要该设备尚未被攻陷至会拒绝停用指令的程度。如果已经攻陷到一定程度，设备可能不会遵守停用指令。

因此，服务生态系统中运行的后端系统必须令终端无法与关键服务通信。如果设备试图与联网对等体或关键服务进行交互，后端系统应发出报警，告知管理人员已经发生异常事件。

8.10.1 风险

不实施停用和废弃会带来多方面的风险，从攻击者完全接管整个网络，到允许已被攻陷的设备继续使用联网服务。最常见的风险来自已经终止订阅物联网服务供应商服务的用户。如果没有从网络中停用这些用户，他们可能可以继续与物联网终端网络中的其他对等体进行通信，或者访问其本不应继续访问的服务。这会增加物联网服务供应商的成本，因为供应商必须为服务生态系统的带宽、CPU 时间和存储容量付费。

8.11 未授权的元数据收集

现代物联网旨在连接现实世界和数字世界。在这个现代化模型中，技术的影响远比以往更加深入。企业或个人可以利用元数据有意追踪和监控随机或特定消费者的行为。

当两个网络实体之间的通信被加密，但是确定消息类型或发送者和/或接受者身份的协议结构已经暴露时，会使用元数据分析。这些元数据可用于推测意图。

假设这样一种情况，汽车发出的消息中包含属于特定消费者的元数据。任何人只要能够（本地或远程）追踪这些元数据，就可能可以监控消费者的动作，然后从这些动作推测行为或意图。

如果汽车的远程信息处理系统中存在可利用的安全漏洞，就可能可以追踪并针对特定消费者的远程信息处理系统，危及其人身安全。

法律组织和保险公司都十分关注这些风险对未来汽车金融的影响，并且开始参与制定相关法律和标准，以规定工程师设计远程信息处理设备的方式。随着更多技术的开发，这一变化将最终扩散至不太活跃的物联网垂直行业。

为了应对元数据收集，应当尽可能多地对数据进行加密，并为通信模块采用唯一的二进制标识符。实施策略以禁止外部用户使用物联网系统 API 从用户配置文件推测硬件序列号以及其他可追踪身份。尽可能禁止将消息结构透露给第三方。禁止将操作、活动或行为透露给第三方。确保所有用户隐私相关数据的保密性和完整性。

8.11.1 风险

使用弱通信安全可能会放任会危及最终用户或暴露最终用户隐私的数据或元数据收集。由于保险公司正在就技术实施最终用户安全要求进行立案，如果企业不对其设备生成的数据负责，可能会招致风险。

9 低优先级建议

低优先级建议是一系列适用于应对成本极高或不太可能影响终端设计的风险的建议。尽管这些建议很有价值，且建议内详述的信息非常重要，但所讨论的缓解或修补策略可能超出了特定企业的范围。请评估每条建议，确定所描述风险是否与企业及其客户相关或有重要关系。如果客户要求解决这些风险，请应用这些建议。

9.1 有意和无意拒绝服务

无线电通信始终面临的一个威胁是*干扰*，或是故意广播可用于扰乱合法信号的噪音或图形。由于无线信号仅仅是由按照特定图形在空间中飞行的电子构成，很容易就可以编制一系列信号来中断或扰乱形成通信数据的图形。

此类攻击的目标通常只是为了中断信号，从而禁止或阻止合法用户获取服务。在另外一些情况下，攻击可能更具针对性。例如，可以*欺骗*没有身份验证机制的通信协议。为此，必须对真实信号进行*干扰*，以使攻击者的欺骗信号更有可能抵达目标。

例如欺骗全球定位系统 (GPS)。民用 GPS 信号缺乏加密和身份验证，因为它本质上是任何人都能够接收的明文广播信号。它还是一个相对较弱的广播信号，很容易因为环境异常而衰减，例如电视接收器和微波炉使用的超高频 (UHF) 预放大器。

对于需要本地信息才能正常工作的设备，GPS 信号受到干扰可能会形成可靠性风险，继而逐级演进成为信息安全风险，尤其是之后有人实施欺骗的情况下。

为了应对干扰和其他形式的故意拒绝服务 (DoS) 攻击，可制定强有力的通信协议，着重降低服务中断的影响。网络应当检测设备是否突然或以异常方式离开网络。每个终端在准备离开网络时都应当“道别”。如果设备没有道别，应当记录该异常情况以便统计分析。

此外，设备每次重新加入网络时，都应当重新协商通信安全密钥。不应使用相同的通信安全密钥。应当用相同的非对称密钥*引导加载*，但对于每个通信会话，密码协商都应当得出新的对称密钥。

非故意的无线电干扰有很多原因：环境条件阻止信号传播，设备故障，甚至是工作频率相同的相邻设备。无论根本原因是什么，依赖无线电通信的工程师都会遇到导致信号恶化或损耗的暂时性条件。这些损耗必须通过应用程序*和*网络通信协议的设计进行补偿。

我们建议开发人员阅读 GSMA 连接效率指南 [9]，其中包括如何防止非故意拒绝服务攻击的建议，并且提供了关于设备主机身份报告 (DHIR) 的指导。

9.1.1 风险

如果未能应对故意拒绝服务攻击的风险，会导致异常或不安全的终端行为。如果终端始终使用同一个会话密钥，攻击者可能通过这种方式滥用网络架构，收集用于保障通信安全的对称密钥的相关信息。在每个会话断开之后正确建立安全会话，对于终端通信安全是十分必要的。

9.2 安全关键分析

大多数物联网产品会将现实世界的某些方面与数字技术相结合。因此，人们很可能根据物联网终端提供的信息在现实世界中作出决定。此外，物联网终端通过从数字世界获得的信息作出的决定可能会影响现实世界。

因此，物联网服务供应商必须从安全角度评估其产品，确定人类生命是否，如何以及何时会受到该技术的影响。如果没有采用足够的安全措施来确保技术不会遭到滥用导致人身危害，客户可能会面临风险。

为了解决安全问题，请与物联网服务供应商的管理、法律及保险团队进行讨论。确保这些团队了解产品或服务使用技术的功能和限制。判断这些技术是否能够满足业务需求，并为客户提供将要使用的应用程序所需的安全级别。

9.2.1 风险

显然，如果没有花时间评估产品或服务对客户安全的影响，可能会导致收入损失、意外事故甚至丧失生命。

9.3 阻止隐藏组件和不可信桥接

在与彼此或中央处理单元通信时，物理电路组件通常不会使用任何类似的保密性和完整性。因此，任何攻击者都可以读取或写入这些总线传输的数据。通信安全中的这种漏洞会导致攻击者能够冒充物理电路上的合法设备。如果攻击者愿意，他们可以冒充 NVRAM、RAM 等关键组件，甚至是信任锚。

这种攻击的目的是绕过总线上两个组件之间的安全措施。这种情况的一个典型例子是利用此漏洞绕过分析 NVRAM 中存储的应用程序映像时的完整性验证过程。在 CPU 检索 NVRAM 中存储的记忆内容时，攻击者可以使用直通系统向 CPU 提供真实记忆内容。如果 CPU 上运行的应用程序已经验证了应用程序映像的完整性，攻击者可能会控制物理总线上的通信，有选择地交换

对攻击者有利的 NVRAM 内容。换句话说，CPU 先验证一个应用程序映像（原始映像），然后将攻击者的映像载入 RAM 并予以执行。

防御这种攻击的一种方式是：

- 将 NVRAM 内容载入 RAM
- 验证载入 RAM 的应用程序映像
- 直接在 RAM 中执行代码或将内容缓存在 RAM 中

这里还需要注意的是，攻击者同样可能会破坏 RAM，削弱这一过程的作用。但是，比起攻击 NVRAM，对 RAM 进行中间人攻击要复杂和昂贵得多，因为总线和访问模式的速度比 NVRAM 快得多而且不稳定，NVRAM 主要是按块进行访问。

此外，攻击者可以为已验证 NVRAM 内容的小块区域生成校验和，定期检查 NVRAM 的签名。如果校验和不一致，说明内容受到操纵。这种方法可能成功，但是成功率较低，因为攻击者可能只操纵了运行中的程序没有随机检测到的少量数据。

应当注意，虽然这种攻击的最佳防御方式是验证 NVRAM 内容然后将其载入可执行 RAM，但并没有能够彻底解决这一问题的方案。保护物理组件的成本非常高昂，因此更彻底地解决这一攻击并不现实，除非客户要求这种程度的安全保证。

如果使用 I2C 等更基础的物理通信协议，这种攻击会变得更简单。因为 I2C 等本质上是物理广播系统。因此，I2C 总线上的任何组件都可以冒充任何其他组件。这样，如果通信信道没有实施保密性和完整性措施，攻击者就可以冒充总线上的其他设备。如果需要解决这个问题，应在物理总线协议之上使用的应用程序协议中实施保密性和完整性措施。

9.3.1 风险

完全不实施任何解决方案会导致攻击者得以绕过应用程序的完整性检查。这样攻击者就能够使用引导加载程序或 TCB 等更高权限的代码攻陷正在执行的应用程序。

但需要注意的是，这种攻击的可能性远远低于针对引导加载程序的简单攻击。对 NVRAM 或 RAM 等高速组件进行中间人硬件攻击非常困难和复杂，就目前而言成本也很高。虽然攻击者永远可能通过这种方式破坏嵌入式系统，但成本可能太过昂贵。

因此，将代码载入 RAM 并验证完整性可能是一种比较合理的解决方案，可以绕过大多数攻击。

此外，由于上述及其他原因，不应将密钥保存在这些不安全的权限中。应当将其存储在信任锚中，由 TCB 使用，而不是存储在 NVRAM 等可能被冒充或攻陷的介质中。

9.4 阻止冷启动攻击

冷启动攻击［参考文献］是一种针对计算机系统的物理攻击策略，通过取出计算机的物理存储器并将其放入攻击者控制的另一个系统中，从正在运行的计算机中析取加密信息。这种攻击的好处是，攻击者可以运行自定义的操作系统，将 RAM 内容全部转储到永久存储器中。这样，

攻击者就可以对收集到的数据进行梳理分析，判断是否存在可以利用的安全相关凭证。具体可能包括：

- 加密信息或私钥
- 登录凭据（用户名和密码）
- 个人验证信息 (PII)
- Web 服务的访问凭证

这种攻击的目的是攻陷加密信息，使攻击者得以长期访问本来无法访问的资源。例如，普通攻击者不可能破解最新 TLS 标准中使用的加密算法。但是，如果能攻陷相互验证 TLS 服务中使用的私有客户端证书，攻击者就能利用更为便利的系统模拟客户端。

为了成功实施这种攻击，攻击者必须能够取出目标计算机系统的 RAM，同时不改变芯片中存储的位。根据研究论文中的详细描述，这可以通过冷却存储器芯片来实现。但是，RAM 必须易于取出。如果 RAM 焊接在电路板上，就会大大增加攻击的复杂性，攻击者需要使用焊枪才能取出存储器，可能会导致内容受损。

此外还必须注意，在关机时擦除存储器会有帮助，我们建议采取这种措施以增强终端隐私。但是，冷启动攻击随时都可能发生，即使是在系统运行的时候。因此，擦除存储器可能会有帮助，但可能无法成功阻止现实世界中的攻击。

针对这种攻击，更有效的缓解措施是使用 CPU 内部 RAM 处理安全相关操作。许多 CPU、MCU 和 MPU 都有一小部分内部 SRAM 可供运行中的应用程序使用。如果应用程序将关键安全凭证（如私钥等）的使用限制在该内部 RAM 的范围内，那么可移动（或外部）RAM 内容对于攻击者的价值就会降低。

9.4.1 风险

如果没有考虑冷启动攻击的风险，可能导致攻击者利用简单的攻击模式析取关键安全密钥。如果物联网服务供应商生态系统中的所有终端均使用相同的安全密钥，就可能被大规模攻陷。

有关详细信息，请参考 – https://citp.princeton.edu/research/memory/

9.5 不明显的安全风险（穿墙透视）

尽管通信网络中启用和实施了相互验证、保密性和完整性措施，但传输模式可能与事件直接相关。响应某些物理事件进行数据传输时，最终可以在物理事件和数据之间建立关联。这样，攻击者可能得以监控信号模式，然后从模式中推出含义，无论攻击者是否能够直接访问明文数据。

例如家庭自动化技术，根据用户在特定房间中的实际存在情况作出反应。如果攻击者能够远程监控通信系统，可能仅仅通过观察物联网终端、网关和后端系统之间的通信，了解特定家庭中的用户数量，用户在家中所处的位置，用户的身份。

攻击者可能可以轻松区分人多的家庭，只有一个人独自在家的家庭，以及这个人在家中的位置。保险公司和法律实体需要了解这会如何增加房主和其他租户在居住空间中的风险。

这一风险可能很难应对。最常用也是最简单的防御模型是按照预先定义的速率发送样本，无论是否存在可以采样的用户。如果实施保密性和完整性措施，禁止远程攻击者评估数据的纯文本，观察者就无法区分包含用户活动的样本和空样本。

但是，这一模型存在一些问题，例如提高频谱饱和度，增加低功率或电池供电技术的功耗，解密、验证和解析空样本数据包需要增加处理级别等。

另一种方法是按随机间隔发送有突然变化的样本。这种模式成本较低，功耗需求较低，需要的处理能力也较低。但是，仍然有可能观察到表示用户在场情况的细微变化。例如，任何一个真正的熵系统都是完全随机和不可预测的。然而，用户行为是完全可以预测的。如果用户进入房间，该房间中的传感器会作出反应，开始向网络中的对等物联网终端发送数据，*引入*一致行为可能表示用户在场。

如果任何团队开发的技术面临这类风险，就应当研究暴露隐私可能带来的后果，并咨询法律团队以确定该技术是否会对企业的法律立场或保险模式产生影响。

9.5.1 风险

如果物联网服务供应商没有从隐私暴露和安全风险的角度对自己的技术进行评估，可能需要对架构进行大规模修整才能补偿*必须*解决的风险。应当在工程阶段开始时，或是尽可能早地将这些解决方案设计到产品中，而不是后面再尝试对架构进行代价高昂的调整。

9.6 应对聚焦离子束和 X 射线

聚焦光束 (FIB) 是一种评估半导体常用的生产仪器。这项技术能够对电路进行纳米级别的检查和修改，让分析人员能够识别生产中的故障，并在改变制造过程之前对电路修补方案进行测试。

在信息安全领域中，可以使用 FIB 分接内部总线，以便拦截通过内部组件传输的数据。此外，还可以使用 FIB 改变内部电路，从而改变内部组件的工作方式，让攻击者得以绕过安全限制措施。

几乎所有设备都可能受到 FIB 的攻击。但是，FIB 过程只会针对特定设备进行。这是因为 FIB 本身是一种极其昂贵的技术，一台设备大约需要花费一百万美元。由于技术成本高昂，极少有组织拥有这样的设备。此外，这种设备并不能自动运行。它需要很高超的技术才能操纵，并且需要极高水平的半导体分析专业知识才能利用。因此，FIB 的实际成本远远高于一百万美元这个数字，其本身以及使用者的教育水平、薪资和专业知识会让成本增加至数百万美元。

但是，组织可以进行外包。因为逆向工程在很大程度上是合法的，所以组织会为有兴趣对设备进行逆向工程的客户提供半导体攻击服务。这些业务的费用可能从一万美元到一百万美元，具体取决于攻击特定组件所需的定制化水平和专业知识。例如，外包公司会有一套*固定方案*来绕过普通芯片的保护措施。但是，配备新型安全锁定技术的定制 FPGA 解决方案成本要高得多，因为没有现成的*固定方案*。为了成功实施 FIB，需要采取新的流程，耗费大量时间和金钱。

一些新技术宣称可以防御 FIB 探测，例如现代的信任锚变体。虽然这些宣称具有一定的有效性，但是任何非动态的硬件保护措施（大部分硬件保护措施是非动态的）都会导致攻击者在投入足够时间分析染绕过技术之后编写出一套*固定方案*。因此，这些宣称可能有效，但是只在一*段时间内*有效。

所以，为了补偿此类具有侵略性但几乎总能成功的攻击技术，工程组织设计的安全策略决不能仅仅依赖信任锚来取得成功。相反，必须设计一种方案，使用该技术作为基础信任锚，对每个终端的密钥进行个性化，以免单个设备被攻陷后导致整个终端网络都被攻陷。

假设这样一种情况，攻击者必须使用 FIB 从*每个要攻击的终端*析取加密信息。这很快就会变成一个成本极其高昂的方案，超出几乎任何攻击者的预算。无法*充分缓解*这些攻击方法，就必须*降低其价值*，通过架构而不是隐匿来减轻风险。

9.6.1 风险

FIB 的风险在于可以析取组件中的加密信息和其他知识产权，即使是采取了增强安全措施的组件。由于无法采用低成本的方式阻止 FIB，组织必须改变其保护终端系统的策略，或是承受终端生态系统完全被攻陷的风险。

9.7 考虑供应链安全

任何计算系统的安全都要从组成电路板的原始元件开始保障。嵌入式系统的晶片、加密凭证、只读存储器 (ROM)、固件和其他核心属性都有助于保障此类系统的安全。如果其中任何一个组件遭到篡改，则整个系统的安全都可能遭到破坏。

因此，关注安全问题的物联网服务供应商必须考虑组件的来源、组装过程以及用于交付组装技术的实现过程。如果用于生产技术的过程没有经过仔细规划，过程中的任何一点问题都可能导致严重的安全故障。

请思考以下问题：

- 晶片在何地由何人生产？
- 晶片设计是否经过可信的第三方信息安全团队分析？
- 晶片是否会在安全的设施内制造？
- EEPROM 或 NVRAM 会如何填充可执行映像，例如引导加载程序？
- 闪存可执行映像的过程是否安全？
- 通过何种方式将可执行映像交付给制造商？
- 可执行映像在闪存到 EEPROM 或 NVRAM 上之后是否会经过验证？
- 通过何种方式将加密信息置备在芯片上？
- 如果加密信息由制造商生成，那么密钥是否使用经过验证的 RNG 生成？

- 是否所有安全密钥都唯一，符合 TCB 建议？
- 密钥通过何种方式共享给物联网服务供应商？是否安全？
- 唯一芯片标识符（序列号等）通过何种方式与密钥关联并共享给物联网服务供应商？

虽然选择更安全的工厂来构建和组装产品可能会增加成本，但这一步对于组织是必不可少的。这取决于产品用例、预期部署环境、预期客户以及其他因素，如人员安全、军事应用和关键基础设施部署等。如果产出技术可能影响到人身安全，就必须评估供应链是否有安全漏洞。

9.7.1 风险

如果不能保障供应链安全，组织会面临许多风险，有一些可能是完全意想不到的，但对于企业而言至关重要：

- 终端复制（非法生产）
- 盗用技术（竞争对手盗用服务供应商的技术，抢走业务）
- 凭据盗窃（数据拦截或冒充攻击）
- 注入攻击（可能会在以后激活的恶意“后门”）

9.8 合法拦截

合法拦截是指合法拦截或操纵客户与服务提供商之间通信的行为，可能是以下两种方式之一。第一种，最典型的情况是执法机构向运营商提交司法请求，要求访问特定用户通信的元数据或实际数据。第二种，执法机构要求物联网服务供应商提供权限以访问特定用户的数据和/或元数据。在执法机构要求通过运营商访问的情况下，可能并不会将相关问题告知物联网服务供应商，具体取决于法律请求的范围。因此，服务供应商必须准备好实施或遵从此类机构提出的司法请求。

所以，供应商必须确定执法请求可能导致哪些隐私问题，并且需要准备好在其法律能力内提供关于组织法律模式和隐私政策的相关信息。

近年来，谷歌、苹果和其他大型机构采用了*告密保证*，在有人代表机构向公司提出秘密请求时合法地通知用户。企业可以删除表示合法拦截机构*没有*与其联系的措辞、图片或其他对象。删除这种对象就表示有机构提出了请求。

9.8.1 风险

如果企业没有为合法拦截请求做好准备，那么在有人提出此类要求时，企业就会处于不利境地。企业可能需要遵从请求，但可能没有准备好法律基础设施或隐私政策，因而会面临潜在风险。

没有为终端协议和物联网平台做好充分准备以保证足够的保密性和完整性，会使得通信在企业不知情的情况下，在网络侧遭到拦截。这样企业可能就会泄露用户数据，或者陷入类似斯诺登 NSA 泄露等事件之中，大大降低公众对组织保护用户数据能力的信任。

10 总结

总而言之，通过明确定义的架构、在安全相关事件发生前或发生过程中识别风险的智能技术，以及处理此类事件的政策和程序，几乎所有物联网产品或服务中的安全风险都能成功应对。通过分析对于物联网服务供应商而言至关重要的高级别安全概念，可审查常见的安全问题。该过程可指导工程团队执行解决其安全架构漏洞最相关的建议。

随着团队不断完善其架构定义，在实施中会碰到越来越独有的安全问题和疑虑，此时可查看各个独立的建议。

整体而言，每个工程团队都会面临极其相似的风险。组织必须选择与同行分享其遇到的问题，以便建立风险和补救策略的公共知识库。大家一起努力，共同构建相关技术和知识，以帮助彼此保护未来物联网的安全。

附录 A 使用通用引导架构的示例

多跳网络的总体安全级别由链条中最弱的一环决定。因此，物联网终端和网关之间的本地链路需要保证与广域网相当的安全级别，以保持相同的总体安全级别。

通用引导架构 (GBA) [17] 是一种可以用于实现这一目标的技术，它可以同时用于身份验证和数据完整性。其基础是预共享密钥，这些密钥用于生成时间限制密钥（凭证），作为身份验证和加密的依据。

身份验证是确定某人或某物是否确实符合其声称身份的过程。物联网空间中运行着数十亿个终端，确定哪些通信行为真实可信是重中之重。用于建立这种信任关系的机制需要满足可扩展和可维护的要求。此外，互联网服务的多样性要求验证机制能够适应各种服务，同时仍然维持共同的基础设施。基于 SIM 的网络身份验证是一种经过时间考验的机制。这种验证基础设施的优点是不仅可以进行身份验证，还具备基于预共享密钥的加密功能。终端数量的爆炸式增长和物联网在全球的扩展限制了 SIM 的使用，原因包括网络漫游和能够从无人值守的终端取出 SIM 的安全漏洞。嵌入式 SIM 等技术的出现为基于预共享密钥的认证提供了实用的基础设施，扩展了当前基于 SIM 的网络身份验证。此外，物联网的增长最有可能以毛细管网络的形式发生（如前文示例配置 2、3、4 所示的 PAN）。这些毛细管网络是连接到网关的终端设备群。这些终端设备大部分会是轻型终端设备（即不含 SIM，也没有蜂窝网络连接）。这些轻型终端设备同样需要身份验证和加密功能。在毛细管网络中，身份验证主要由网关承担，减少了整体网络中基于 SIM 的复杂终端设备的数量。这种身份验证和安全应从网络扩展至终端设备，从而在给定终端设备和物联网服务平台之间建立一个安全通道。

基于 SIM 的身份验证旨在服务单个应用程序，即验证唯一终端设备以便进行网络连接。终端设备有多种服务，每种服务的验证需求都各不相同。需要建立一个框架，将网络身份验证扩展至多个服务。通用引导架构 (GBA) 就是专门为此设计的一个框架。GBA 利用基于 SIM 的基础设施在设备和网络应用功能 (NAF) 之间生成基于时间的共享密钥。GBA 是 3GPP 在 3GPP 规范 TS 33.220 [17] 中标准化的一种身份验证方法。这种方法可以对有 3GPP 服务订阅的设备进行身份验证。订阅凭证在设备中，通常存储在 SIM 上，例如通用集成电路卡 (UICC)，或者作为远程管理凭据在嵌入式 SIM (eUICC) 上进行存储和管理，例如 GSMA 指定的嵌入式 SIM (eUICC) [5]。

该框架的优点包括：

- 基于设备和网络应用功能之间的唯一 PSK 或基于共享密钥的 UE 身份验证和基于证书的 NFA 身份验证的相互验证 (TS 33.222) [18]。
- 可以在受信任的环境中保护凭据。
- 如果使用 eUICC，可以对凭据进行远程更改。
- 可扩展。身份验证“内建”在框架中，因此维护的复杂性和成本会随着设备的数量线性增加。
- 数据完整性。验证期间生成的基于时间的密钥可用于建立 TLS-PSK 隧道，建立这种连接可以提供很强的数据完整性和保密性。

附录 B 关于在物联网服务中使用 UICC 卡的教程

ETSI TS 102 221 中规范的 UICC 是一个智能卡平台（可编程防篡改安全元素），可以为 UICC 托管设备提供可互操作的安全文件系统接口和安全应用程序框架。ETSI TS 102 221 为 UICC 托管设备提供了一个框架，以便发现 UICC 上的相关应用，每个 UICC 应用程序对应于一组已知的置备和配置信息以及操作过程（例如身份验证或密钥导出），可由托管设备根据其需求提供支持。

在物联网环境中，UICC 可按 ETSI TS 102 671 标准的规定提供多种设计规格和工作环境范围。在最简单的实施例中，UICC 通常由网络运营商所有，仅托管一个网络接入应用程序（3GPP TS 51.011 规定的 SIM 应用程序、3GPP TS 31.102 规定的 USIM、3GPP2 规定的 CDMA CSIM、WiMAX SIM 等）。在这种情况下，UICC 提供了一个标准化容器，可在移动设备上托管安全置备和配置信息以及加密程序，从而实现网络接入，此外还有附加机制，可以使用 ETSI TS 102 225/TS 102 226 远程管理 UICC 内容。移动网络生态系统具有相应程序，可以确保 UICC 在网络运营商的控制下安全进行个性化和部署，从而在 UICC 托管设备和基础设施之间建立单独的共享对称密钥。

UICC 平台的一个重要特性是支持孤立安全域，让复杂生态系统中的多个利益相关者可以在 UICC 上分配各自的区域，并避开其他利益相关者，对其内容进行保密管理。该功能通过 ETSI TS 102 226 继承自 GlobalPlatform 卡片规范 [15] 附录 A。因此，在物联网环境中，一个 UICC 就能让多个利益相关者彼此独立地存储和管理各自的凭据。

UICC 一般可以承载几个网络接入应用程序（任何给定时间只能有一个处于活动状态），还可能可以承载其他应用程序以保护对更完备服务的访问，例如用于 IMS 访问的 ISIM 应用程序（如 3GPP TS 31.103 所述），或者对于物联网服务而言，承载 oneM2M TS-0003 附录 D 中所述的 1M2M SM 应用程序。1M2MSM 应用程序可支持直接置备专用物联网服务/应用程序凭据，以及使用 3GPP 规定的 GBA 机制从 UICC 上预先存在的网络访问凭据导出的凭据。它进一步使物联网服务供应商能够根据其特定需求定制加密程序，例如支持特定服务验证机制。

一个 UICC 还可以承载多个 1M2MSM 应用程序，实现各个物联网服务供应商专用对称密钥的机密部署。UICC 所有者（在物联网环境中通常是网络运营商或 OEM 制造商）可以在 UICC 上与提出请求的物联网服务供应商共享空间，这样物联网服务供应商也可以利用启用网络访问凭据安全部署的认证 UICC 个性化链和基础设施来部署自己的凭据。

在物联网应用程序安全依赖于非对称加密的情况下，可以类似地使用定制 UICC 应用程序，根据特定物联网服务的需求帮助部署公钥/私钥对。此类 UICC 应用程序需要根据物联网应用程序的特定基础在托管设备上指定和支持。

附录 C 文档管理

A.1 文档历史

版本	日期	变化的简要描述	授权批准	编辑/公司
1.0	2016 年 2 月 8 日	新版 PRD CLP.13	PSMC	Ian Smith GSMA

				和 Don A. Bailey Lab Mouse Security
1.1	2016 年 11 月 07 日	新增 GSMA 物联网安全评估计划参考资料。 少量编辑解释说明。	PSMC	Ian Smith GSMA
2.0	2017 年 9 月 29 日	新增 GSMA LPWA 网络资源参考并增加少量更新。	物联网安全团队	Rob Childs GSMA

A.2 其他信息

类型	描述
文件所有者	GSMA 物联网项目
联系人	Rob Childs – GSMA

为您提供卓越的产品是我们不懈的追求。如果您发现任何错误或遗漏，请告诉我们。您可发送邮件至 prd@gsma.com

欢迎您随时向我们提出建议和问题。