AN12385

使用 NXP 安全 MCU LPC54S0xx 构建安全系统

版本 0 — 2019 年 3 月 15 日

应用笔记

目录

1 介绍

在介绍安全 MCU LPC54Sxx 之前，本章介绍了有关嵌入式系统安全的基本知识。

1.1 什么是安全

本文档中的术语“安全”是指保护计算机系统免遭盗窃或损坏硬件、软件或电子数据。它还包括服务中断或误导。由于对计算机系统，互联网和诸如蓝牙/WiFi之类的无线网络的依赖性增加以及“智能”设备的增长，该领域的重要性日益提高。

1.2 基本安全原则

机密性（Confidentiality），完整性（Integrity）和可用性（Availability），也称为 CIA 三位一体，是一种旨在指导安全策略的模型。在任何类型的安全系统中都应保证这三个关键原则。如果违反了三者中的任何一项，可能会对有关各方造成严重后果。

- **机密性**

  机密性是隐藏信息以防止未授权访问的能力。在安全方面，这可能是 CIA 三位一体最明显的方面。但是相应地，它也是最容易受到攻击的。加密和加密方法是尝试确保从一台计算机传输到另一台计算机的数据的机密性的示例。

- **完整性**

  完整性是确保数据是原始安全信息的准确且不变的表示形式的能力。一种安全攻击是在将一些重要数据发送到预期的接收者之前，对其进行拦截并对进行更改。

- **可用性**

  可用性是确保所需的信息随时可以被授权的查看者访问。某些类型的安全攻击企图拒绝对适当用户的访问，或者为了给他们带来不便，或者是因为存在一些其他次要影响。例如，通过为特定搜索引擎破坏网站，竞争对手可能会变得受欢迎。

1.3 威胁和一般的保护原则

我们可以根据攻击类型和位置来描述威胁。攻击类型包括物理攻击和逻辑攻击。

- **物理攻击**：

  攻击者可以通过直接操纵或观察设备的操作来利用设备中的漏洞进行攻击。

- **逻辑攻击**：

  攻击者仅依靠发送到设备的消息造成损害。

  基于位置的攻击包括本地攻击和远程攻击。

- **本地攻击**：

  需要物理访问设备，这意味着它们不可扩展，并且攻击者需要更多技能。但是，有可能通过局部攻击获得产品的整个代码。分析该代码可能会发现设备漏洞，这些漏洞可能会通过远程攻击应用于类似设备。

- **远程攻击**：

可以通过网络连接发送命令来执行。不需要攻击者在目标设备附近操作。由于其可伸缩性，这些攻击是最危险的。

“物理”和“逻辑”攻击都可以在本地和远程执行。图 1 的左侧显示了具有每种类别典型攻击的威胁范围。 类别有重叠。大多数远程攻击确实是逻辑攻击。但是，rowhammer 攻击是远程物理攻击的一个示例。Rowhammer 攻击通过反复更改内存位置的内容以导致相邻的远程访问内存块发生更改来攻击目标设备。攻击者可能会影响使用该内存的程序的执行。

图 1 的右侧为以下问题提供了指导：应该防止哪些攻击？该图总结了要防御的攻击及其相对优先级。但是，应在风险和保护的成本之间进行权衡。

防范远程攻击是最重要的。无论是逻辑上还是物理上，都需要保护 IoT 设备和后端服务免受远程攻击。但是，抵御本地攻击是另一回事：如果攻击者可以本地访问设备，则防范逻辑攻击将是下一个优先事项，因为它可以再次由非专业人员自动执行。本地攻击也可能在实际发起攻击的人不知情的情况下发生。例如，当恶意软件侵犯智能手机或 USB 盘时。

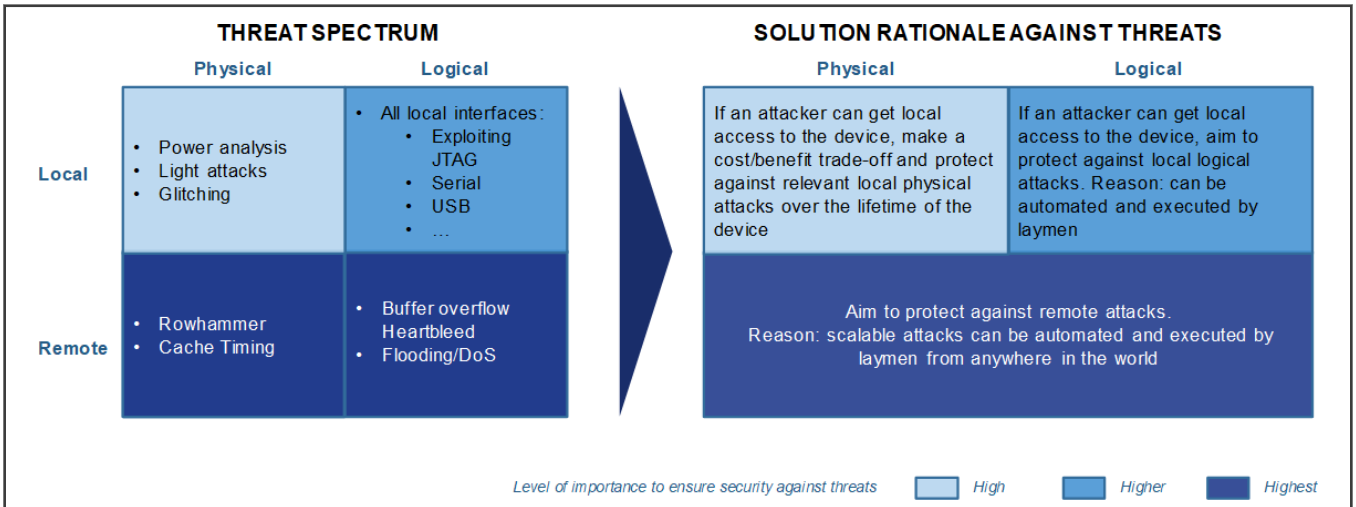

图 1. 威胁范围和解决方案

为了保护物联网数据，必须满足四个基本要求：必须注意完整性和真实性，并在需要时保证机密性和可用性。请参阅 图 2。这些要求听起来很简单，但是，当攻击者主动将设备作为目标时，很难达到所有这些要求。例子有很多：恶意软件注入会导致完整性丧失；伪造设备或克隆破坏了真实性；在被黑客入侵的玩具的示例中，机密性受到破坏，在 DDOS 攻击中，可用性被破坏。

这些要求可以映射为物联网解决方案的四项原则：

- 防止对物联网设备的攻击
- 能够在设备遭到破坏后恢复
- 降低受感染设备的吸引力
- 安全的端到端通信

设计设备时，关键原则是要防止设备受到攻击。但是，请记住，没有绝对的安全性，每个设备最终都可能被黑客入侵。 更高的安全级别只会增加黑客入侵设备的工作量。在设计设备时，应考虑到设备在其生命周期内可能会被黑客入侵，因此您应该能够在设备受到威胁后恢复。这要求您即使在设备受损后也可以始终获得信任的根源。

要评估所需的安全级别，请评估黑客入侵设备的成本与黑客成功后获得的潜在价值相比。只要前者胜过后者，该设备就不太可能成为黑客的主要攻击目标，从而降低了该设备受到攻击的吸引力。不幸的是，攻击者的商业诉求对于设备的开发人员而言并不总是显而易见的。

基于上述物联网解决方案的原理，得出了对物联网设备的以下要求：

- 安全的系统生命周期：系统应该能够安全地度过其不同的生命周期，包括加电/引导阶段，调试，固件/软件的 OTA 更新以及停用。
- 加密和密钥保护：具有私钥的设备可以进行身份验证并与受加密保护的其他设备通信。安全存储此私钥以防止盗窃或提取密钥至关重要。例如，加密和密钥保护实现了安全通道，可用于启动和加密传输至云端，以及将敏感数据加密存储在芯片的内存中。

- 资源隔离（硬件和软件）：资源隔离允许对数据，进程和外围设备的访问和操作进行更严格的控制。带 TrustZone 的 Arm MCU 是资源隔离的一个很好的例子。
- 运行时完整性和认证：由于 IoT 设备通常连接到云并且没有人机界面，因此可以远程评估和认证设备的运行时完整性非常重要。

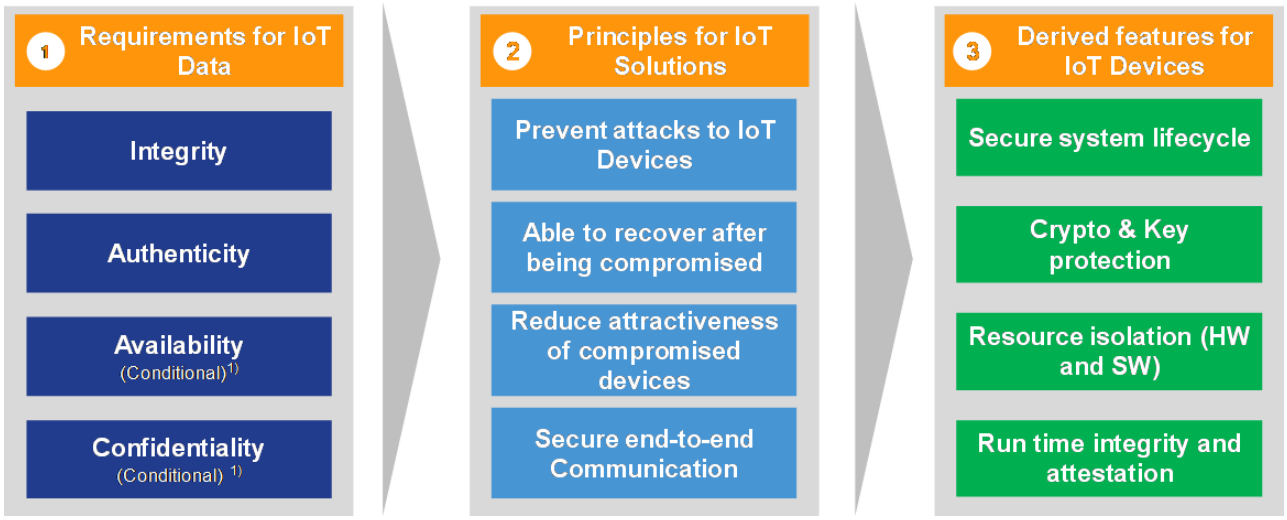

图 2. 为满足 IoT 数据要求而推出的 IoT 设备安全功能

1.4 安全解决方案的局限性

所有安全解决方案旨在仅防御它们可能遭受攻击的部分。 防御所有可能的攻击是一项不可能的任务。如果对某人“有价值”，那么总有人愿意花时间和金钱使用所有可能的方法来破坏安全系统。因此，设计必须确定要保护的资产，以及要保护资产的哪些可能的攻击。这也许是设计过程中最关键的部分。如果一次成功的攻击花费的时间太长或花费金钱太多，那么防御就是成功的.

2 LPC54S0xx 安全架构

本章介绍安全 MCU LPC54S0xx 的体系结构。LPC54S0xx 上的安全系统具有四个硬件模块和 ROM 代码，以实现设备的安全功能。硬件包括一个 AES 引擎，一个 SHA 引擎，一个随机数生成器和一个密钥存储模块，该模块支持 OTP 中存储的密钥或基于 SRAM 的 PUF（物理不可克隆功能）中的密钥。图 3 显示 LPC54S0xx 安全系统的概述。处理器或 DMA 引擎可以访问系统的所有组件，进而加密或解密数据以及进行哈希处理。ROM 除了提供对各种安全功能的支持外，还负责安全启动。

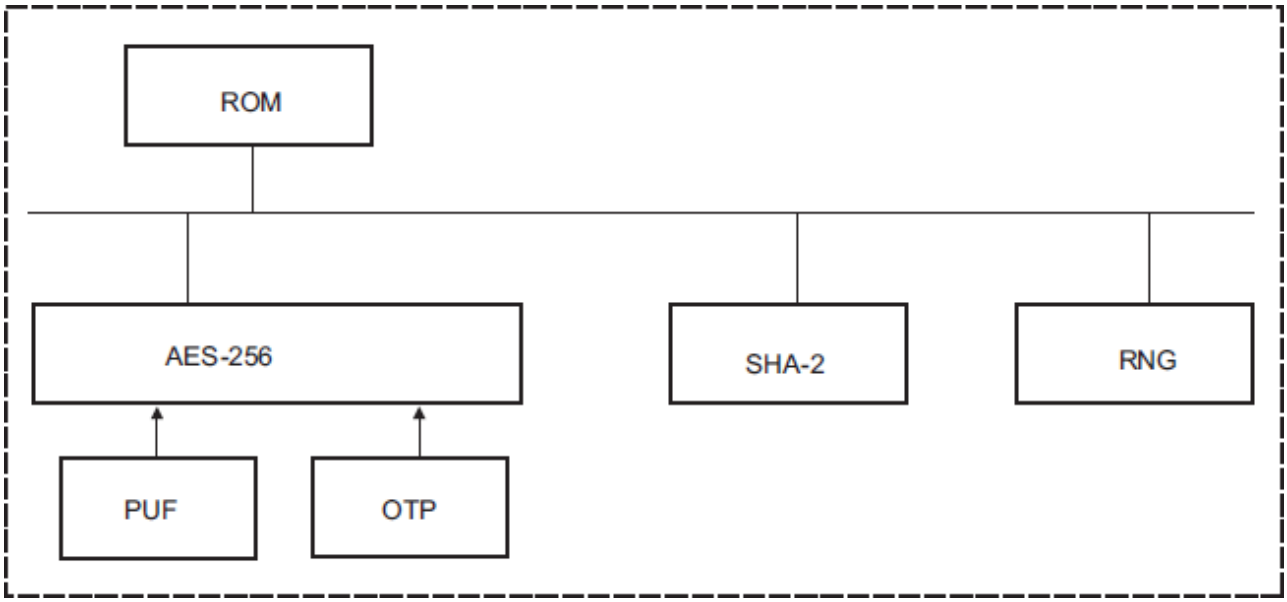

图 3. LPC54S0xx 安全系统

2.1 AES 引擎

LPC54S0xx 器件提供片上硬件 AES 加解密引擎，以保护程序内容。它还加速了数据加密或解密，数据完整性和来源证明的处理。AES 引擎可以使用 OTP 或 PUF 中的加密密钥或软件提供的密钥对数据进行加解密。

AES 引擎的功能特征:

- 数据加解密。
- 安全访问运行时软件无法读取的 AES 密钥（OTP 或 PUF）。
- AES 引擎支持 128、192 或 256 位密钥长度的多种模式：ECB，CBC，CFB，OFB，CTR，GCM。
- AES 引擎符合 FIPS Publication 197，AES。
- 小端模式数据处理。
- 支持 DMA 控制器的数据传输。

2.2 SHA

LPC54S0xx 器件提供片上哈希算法支持，可执行 SHA-1 和 SHA-2 的 SHA-256。哈希是一种将任意大的消息或代码镜像压缩为相对较小的固定大小“唯一”数字（称为摘要）的方法。SHA-1 哈希产生一个 160 位的摘要，而 SHA-256 哈希产生一个 256 位的摘要。建议使用 SHA-256。

对于 SHA 硬件：

- 即使对输入消息进行很小的更改也会导致摘要输出的重大更改。因此，对于给定的输入消息/镜像，只有一个摘要。
- 没有一种可预测的方式来修改一个输入以产生特定的摘要。不能以任何直接方式添加，插入或修改消息以获取相同的哈希值。

这两个属性使它对于验证消息是否有效（不管是有意还是无意）都非常有用。

2.3 RNG

LPC54S0xx 器件提供了生成随机数的片上熵。RNG 生成无法合理预测的 32 位随机数。随机数生成器用于密码，建模和仿真应用，这些应用采用必须以随机方式生成的密钥。

RNG 功能特征：

- 通过 API 调用访问 RNG。
- 遵循 FIPS 140-½, NIST, DieHard。

支持 128 位和 256 位的熵。RNG 产生随机数的随机性的质量依赖于内部逻辑的初始状态。为了构造出 128/256 位的随机数，首先读取 32 位的随机数，接下来读取的 32 位随机数丢弃，再读取 32 位的随机数，以此类推。这样的话，在使用过程中间隔的 32 位随机数被真正使用。典型的应用是，RNG 产生的随机数作为 PRNG 的种子，PRNG 生成的随机数可用于加密算法中。

2.4 OTP

OTP 存储器包含四个存储区（bank），每个存储区可存储 128 位。第一个存储区（OTP 存储区 0）保留使用。其他三个 OTP 存储区是可编程的。在 LPC54S0xx 器件中，OTP 存储区 1 和 2 可用于存储 AES 密钥。OTP 存储区 3 用于客户可编程设备配置数据。

OTP 的功能包括：

- OTP 存储区 3 中储存用户设置数据：
  - ISP 和程序启动源模式
  - 安全启动
  - SPIFI 启动延时
  - 客户定义的位数据
- 信任根（RoT）哈希摘要，用于安全身份验签的引导（OTP 存储区 1、2）
- 加扰的 128 位 AES 密钥，用于安全加密启动（OTP 存储区 2）
- USB 的 VID 和 PID（OTP 存储区 2）
- Boot ROM API 支持，用于对提供 OTP 存储器进行编程

2.5 PUF（物理不可复制的功能）

PUF 是一个新主题，因此让我们对其进行详细讨论。对于 SRAM PUF，这取决于 SRAM 单元上电时的值。从分析上讲，每个 SRAM 单元都有两个表示 1 或 0 的稳定状态。当一个单元加电时，结果状态是无法预测的，但是事实证明，单元中晶体管之间不可控的深亚微米制造差异会导致每个单元优先选择为 0 或 1。对于一组单元，这会导致随机模板（就像是硅指纹），每个 IC 都是唯一的、不可复制的。但是，某些紧密平衡的单元在 SRAM 重启期间可能不稳定，并且会在初始模式 0 和 1（单元翻转）上产生反转的位值。反转后的位数除以模式中的位数即为 SRAM PUF 噪声。由于其嘈杂的行为，SRAM PUF 响应不能直接用作密钥。需要对 PUF 响应进行后处理。这可以通过基于纠错功能和随机性提取器的密钥提取算法来完成。密钥提取器必须能够补偿 PUF 的噪声，并在每次查询时导出相同的加密密钥。参见 图 4，我们可以获得基于硬件的设备唯一密钥。

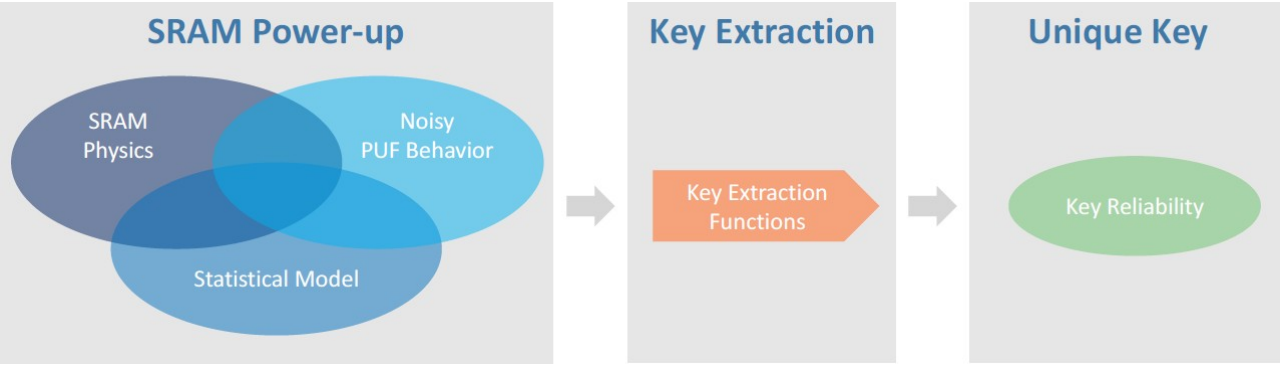

图 4. 独特的密钥生成器

PUF 控制器不仅会生成唯一的密钥，而且还可以在不存储密钥的情况下提供安全的密钥存储。 这是通过使用源自 SRAM 的设备的数字指纹来完成的。代替存储密钥，而是生成密钥代码，该密钥代码与数字指纹结合用于重建要由用户应用程序使用的密钥。

图 5 显示了安全密钥存储系统的顶层框图。框图中的密钥管理还为加密块提供了专用的只读密钥输出接口，其他模块无法读取它。

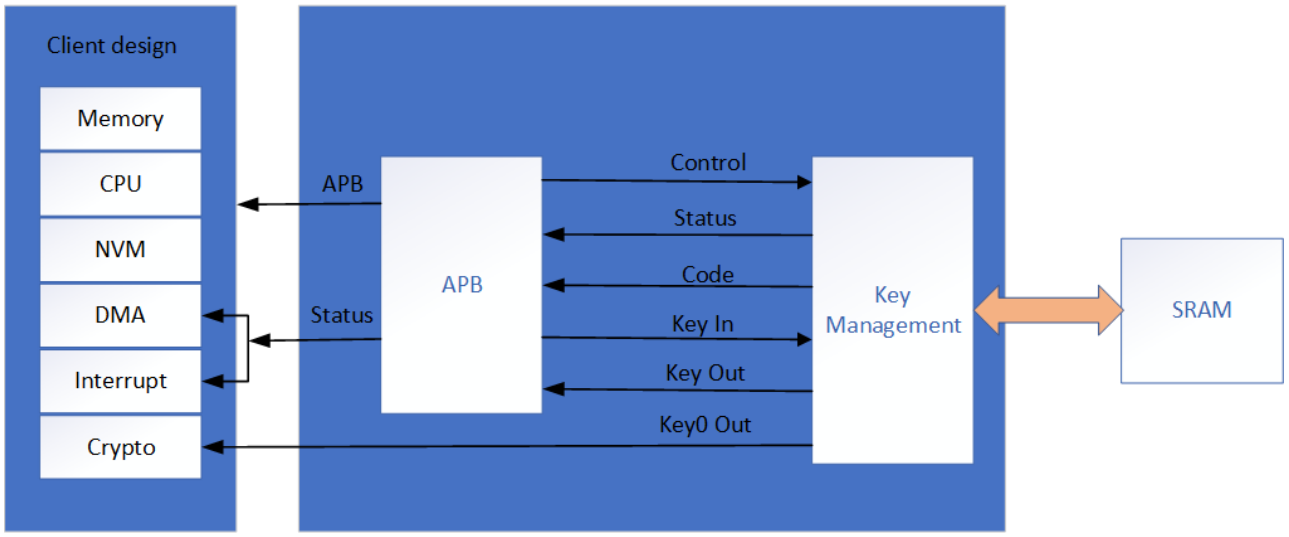

图 5. PUF 框图

2.6 RSA API

LPC54S0xx 设备中没有 RSA 引擎，但是 boot ROM 提供了 API 可以在应用程序代码中执行 RSA2048 签名检查。具有 SHA-256 哈希摘要的 RSASSA-PKCS1-v1_5-SIGN 用于签名验证功能。

Function	Offset	Description
rsa2k_decode_e3	0x00	RSA (2048-bit exponent = 3) decode API Parameter 1: Pointer to secure_param structure Return: 0 => = Success 1 0 => SHA256 comparison failed -1 0 => Not a secure part (Security is not supported by the part) Note: The calling application must ensure that there is at least 1200 Bytes of free space left in the stack. If the amount of stack space left is less, then calling this API might cause a stack overflow.
get_secure_firmware_version	0x04	Get the version number of the security firmware. API takes no parameters. In the return value bit 31 to bit 16 are reserved and will be 0. Bit 15 to bit 8 will contain the major number. Bit 7 to bit 0 will have minor number.

图 6. RSA2048 签名 API

3 LPC54S0xx 安全启动

LPC54Sxx 设备中带有固件（称为启动代码）的 64 KB ROM 存储器。当设备加电或硬件重置时，引导代码必须始终运行。LPC54Sxx 设备中的启动代码支持安全启动，形成信任根（RoT），可以通过防止加载非真实的应用程序代码来保护启动过程。LPC54S0xx 没有用于代码和数据存储的内部闪存 ，因此，镜像文件必须存储在其他位置，以便在复位时加载，或者 CPU 可以从外部存储器（XIP）执行。

安全启动支持将镜像文件从连接到 SPI，SPIFI 和 EMC 接口的外部非易失性存储设备加载到片上 RAM 中，但是安全启动会验证镜像文件，然后决定是否运行它。如果代码是真实的，则将控制权转移给它。建立了从 ROM 到用户代码的可信代码链。安全启动使用以下加密算法：

- SHA256 用于哈希函数。
- 恩智浦定义的“镜像密钥证书”用于公共密钥证书。验证证书中的公钥是否属于 OEM。
- 使用 OTP 内容的 SHA256 哈希检查来验证证书颁发机构的公共密钥或信任密钥的根。
- 具有 SHA-256 哈希摘要的 RSASSA-PKCS1-v1_5-SIGN 用于签名验证功能。使用具有 2048 位公共密钥模数和 32 位公共密钥指数的 RSA 密钥。
- AES 算法是 GCM 模式，用于加密启动。
  - 当将 OTP 用于密钥存储时，将使用 128 位 AES 密钥。
  - 当 PUF 用于密钥存储时，使用 256 位 AES 密钥。

3.1 安全启动镜像文件的类型

为了在运行之前对安全启动代码进行身份验证，对普通用户镜像文件进行签名和/或加密。 安全启动支持以下类型的安全启动镜像文件:

- **签名镜像**：RSA-2048 签名镜像。
- **加密镜像**：AES-GCM 加密和认证的镜像。
- **签名加密镜像**：普通镜像先加密，然后签名。建议使用此镜像。
- **加密签名镜像**：首先对普通镜像签名，然后对包括签名的整个镜像进行加密。

所有安全启动镜像文件都有镜像文件头，该镜像文件头为安全启动提供各种参数，以初始化启动接口，加载地址以及对镜像进行身份验证/解密。

3.1.1 签名镜像

RSA-2048 签名镜像包含以下内容，如 图 7 所示：

- 带有镜像头的镜像文件。
- 镜像密钥证书。
- 整个镜像的 RSA-2048 签名。

在此体系结构中，用于验证用户镜像真实性的方法是在整个用户启动镜像上验证 RSA 签名。用户镜像已使用 RSA 私钥签名。用于签名验证的相应 RSA 公钥包含在已签名用户镜像中的“镜像密钥证书”中。使用具有 SHA-256 哈希摘要的 RSASSA-PKCS1-v1_5-SIGN 算法对镜像进行签名。

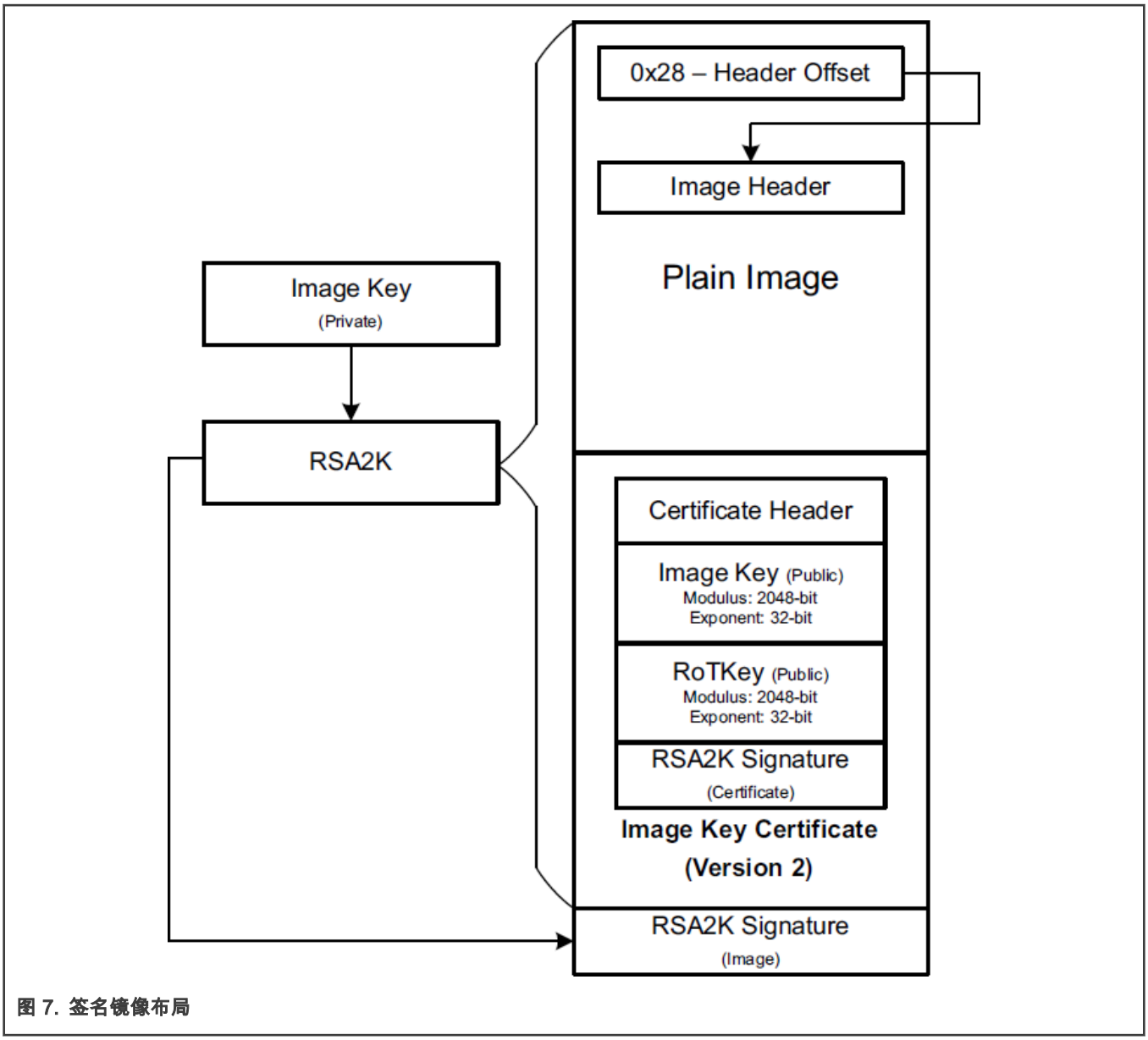

图 7. 签名镜像布局

签名镜像中的镜像密钥证书是什么？在密码学中，公钥证书（也称为数字证书）是用于证明公钥所有权的电子文档。证书由公钥和其他数据（例如人员的姓名和证书到期数据）组成，并使用发布证书的人员或机构的私钥（称为证书颁发机构（CA））签名。如果签名有效，并且软件检查证书的信任 CA，则它可以使用公钥进行进一步的密码学操作。安全启动定义了一种简化的证书格式，称为镜像密钥证书，如 图 8 所示。

签名 = RSASign(RoTKPrivate, SHA256(Certificate_Header || Image_KeyPublic || RoTKPublic))

设备制造商生成的 RoT 密钥用于验证镜像密钥证书，该镜像密钥证书的私钥部分必须妥善保存。密钥公钥部分的 256 位 SHA2 摘要在制造过程中被编程到 OTP 中。

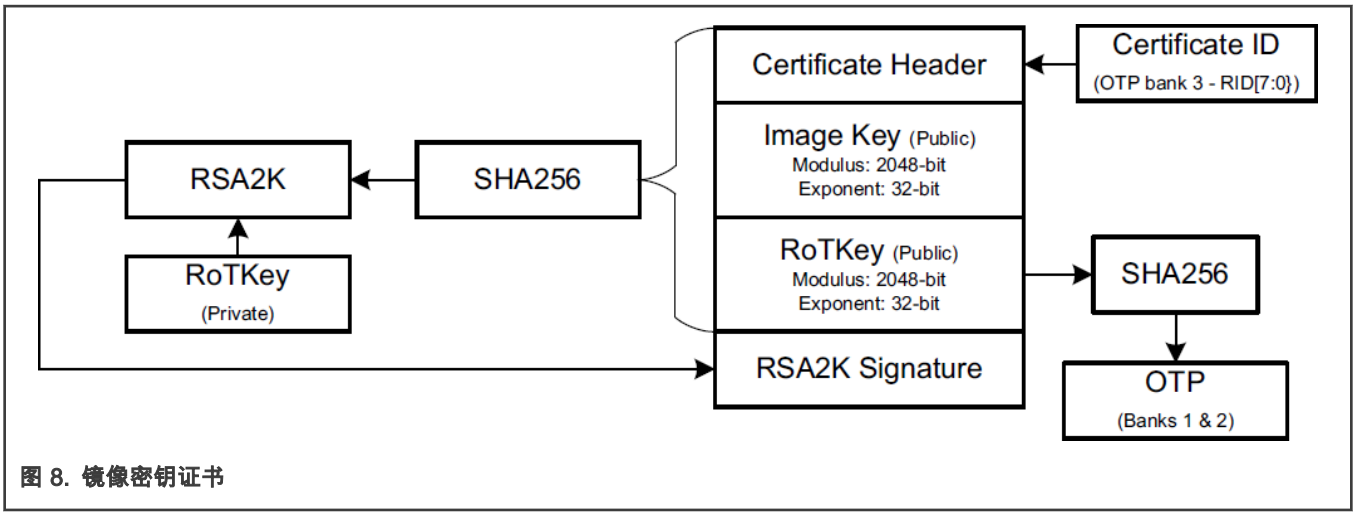

图 8. 镜像密钥证书

3.1.2 加密镜像

加密的镜像包含以下内容，如 图 9 所示。

- 加密镜像
- 加密镜像的镜像头（该头附加在加密镜像的顶部）

加密的镜像文件应通过使用 AES-GCM 算法对镜像进行加密来创建。镜像头附加到 AES 加密镜像的顶部。镜像头包含镜像初始化向量（Image_IV），头初始化向量（Header_IV），加密镜像的身份验证标记（Auth_tag）和头标记（Header_tag）的 GMAC。在加密过程中，“其他身份验证数据（AAD）”为 0。

[Image GMAC tag, Encrypted_image] = AES-GCM(key = OTP_key, IV = Image_IV, Plain_text = Plain_image, AAD = 0)

[Header GMAC tag, 0] = AES-GCM(key = OTP_key, IV = Header_IV, Plain_text = 0, AAD = Encrypted Image Header)

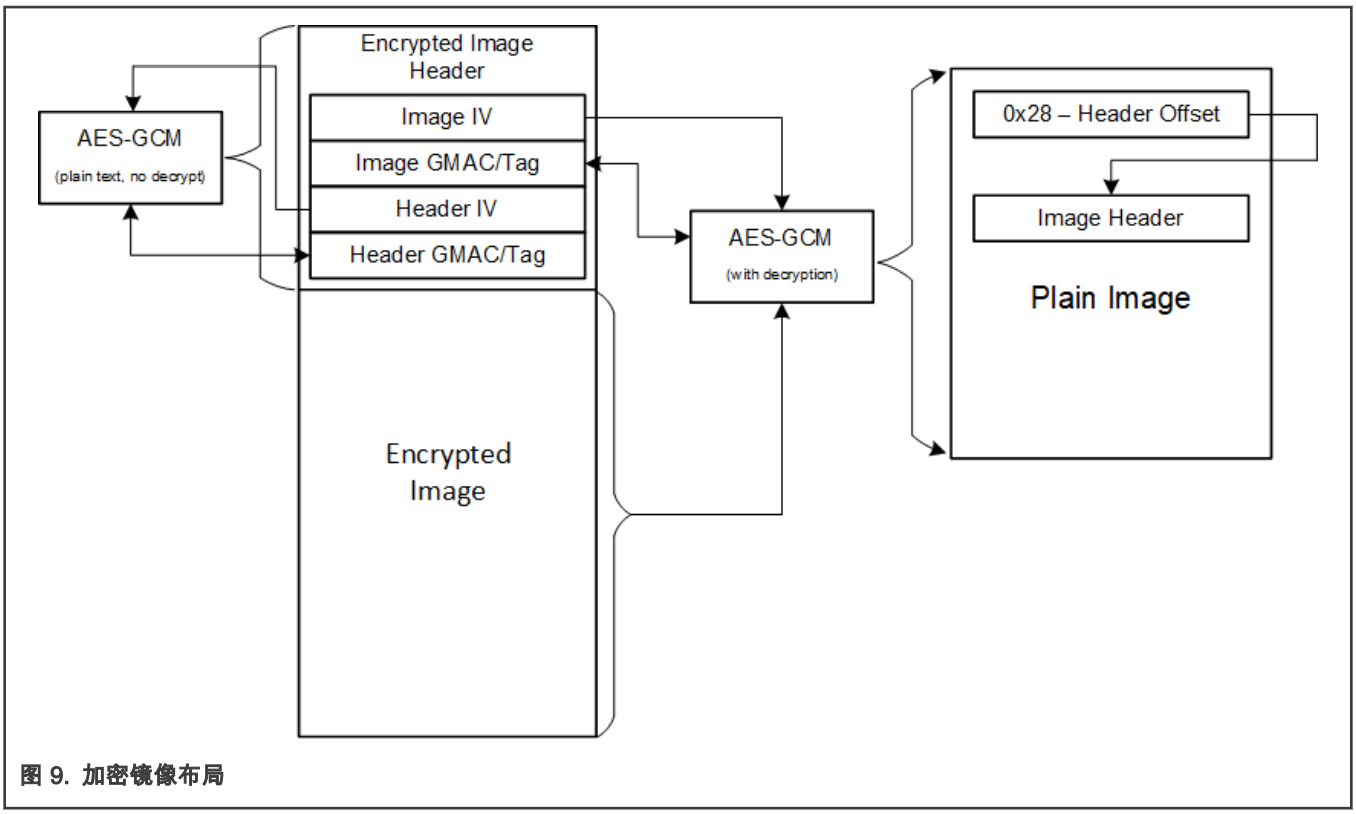

图 9. 加密镜像布局

3.1.3 签名加密镜像

图 10 显示了签名的加密镜像文件布局。普通镜像文件被加密然后签名。要了解有关如何加密镜像的信息，请参阅加密镜像。要了解如何对镜像签名，请参阅签名镜像。

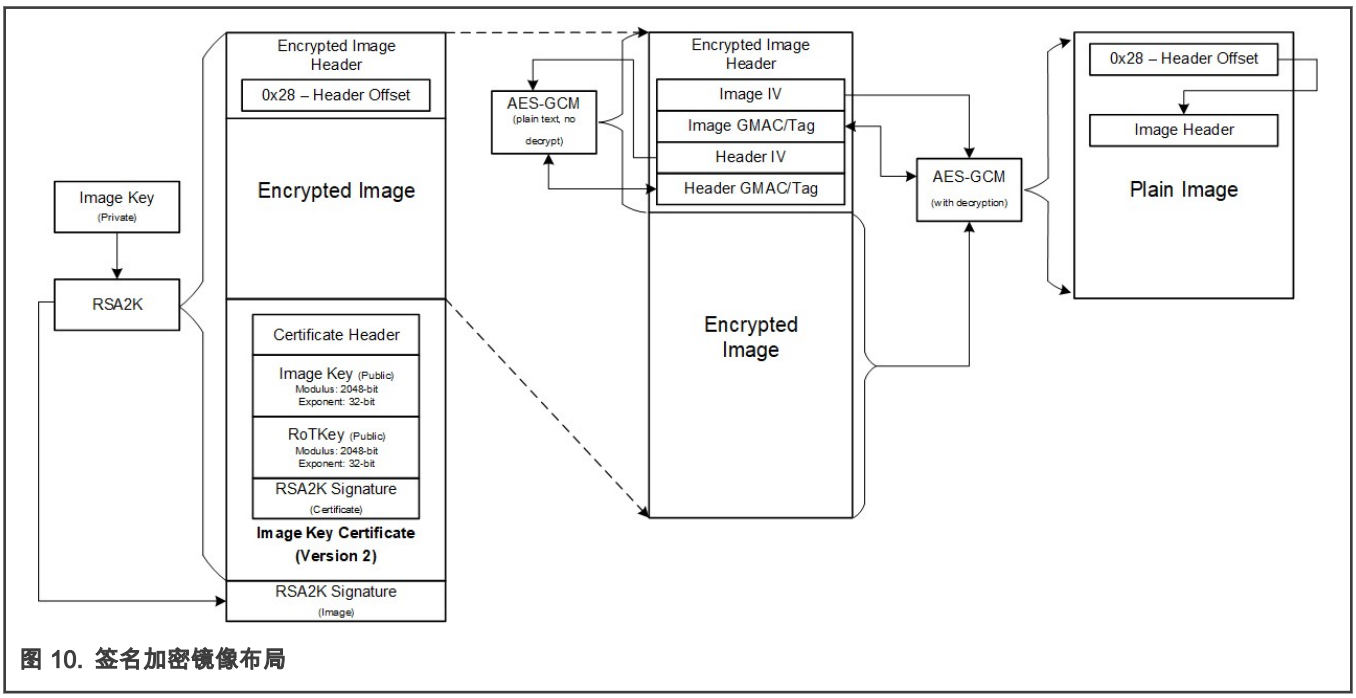

图 10. 签名加密镜像布局

3.1.4 加密签名布局

图 11 显示了加密的签名镜像布局。首先对普通镜像签名，然后对包括签名的整个镜像进行加密。关于如何加密镜像，请参阅加密镜像。关于如何签名镜像，请参阅签名镜像。

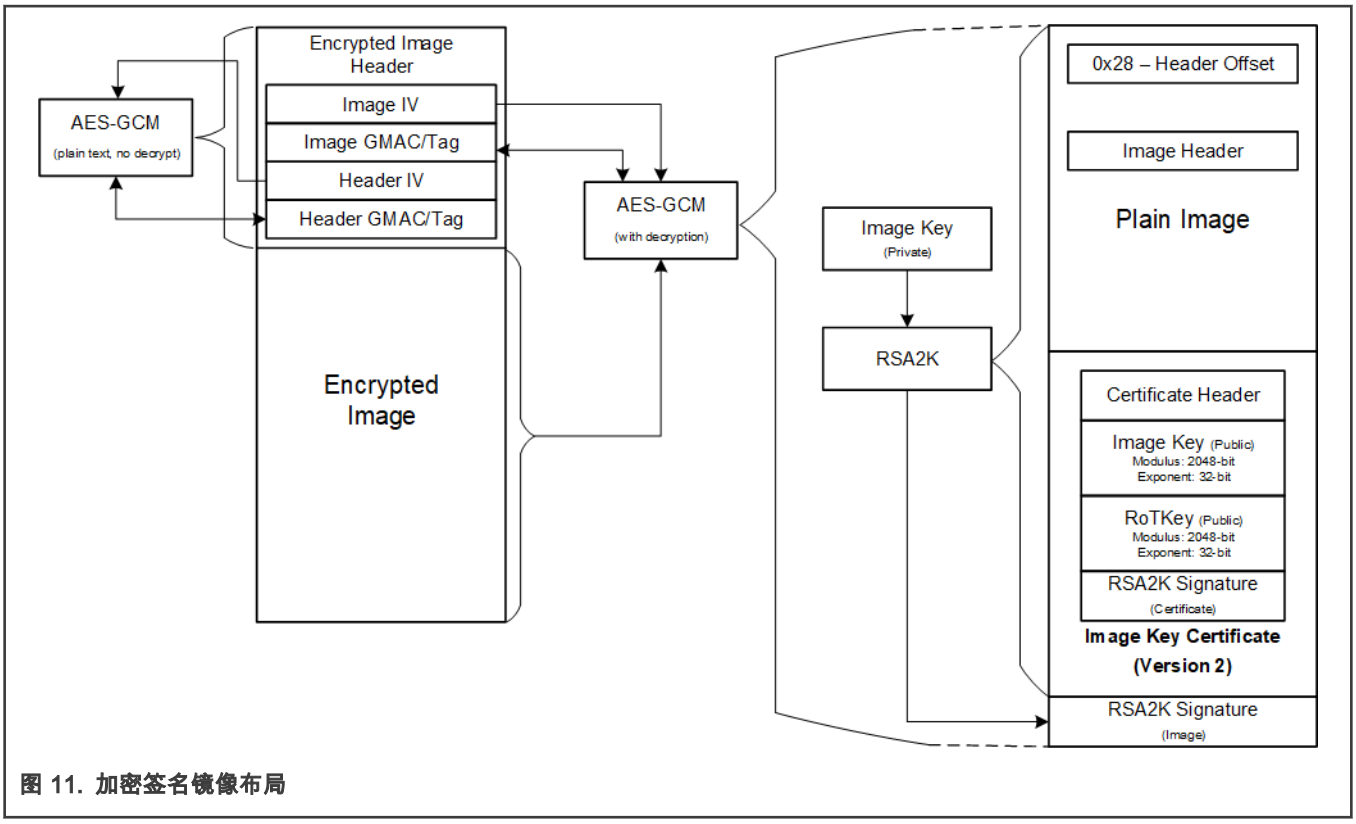

图 11. 加密签名镜像布局

3.2 安全启动过程

每次设备开机或重启时都会执行安全启动，图 12 显示了顶层启动过程。用户代码可以从外部闪存（QSPI 闪存，SPI 闪存和 EMC 闪存）引导，但是需要将其复制到内部 RAM 中，以便在身份验证或/和解密成功后运行。

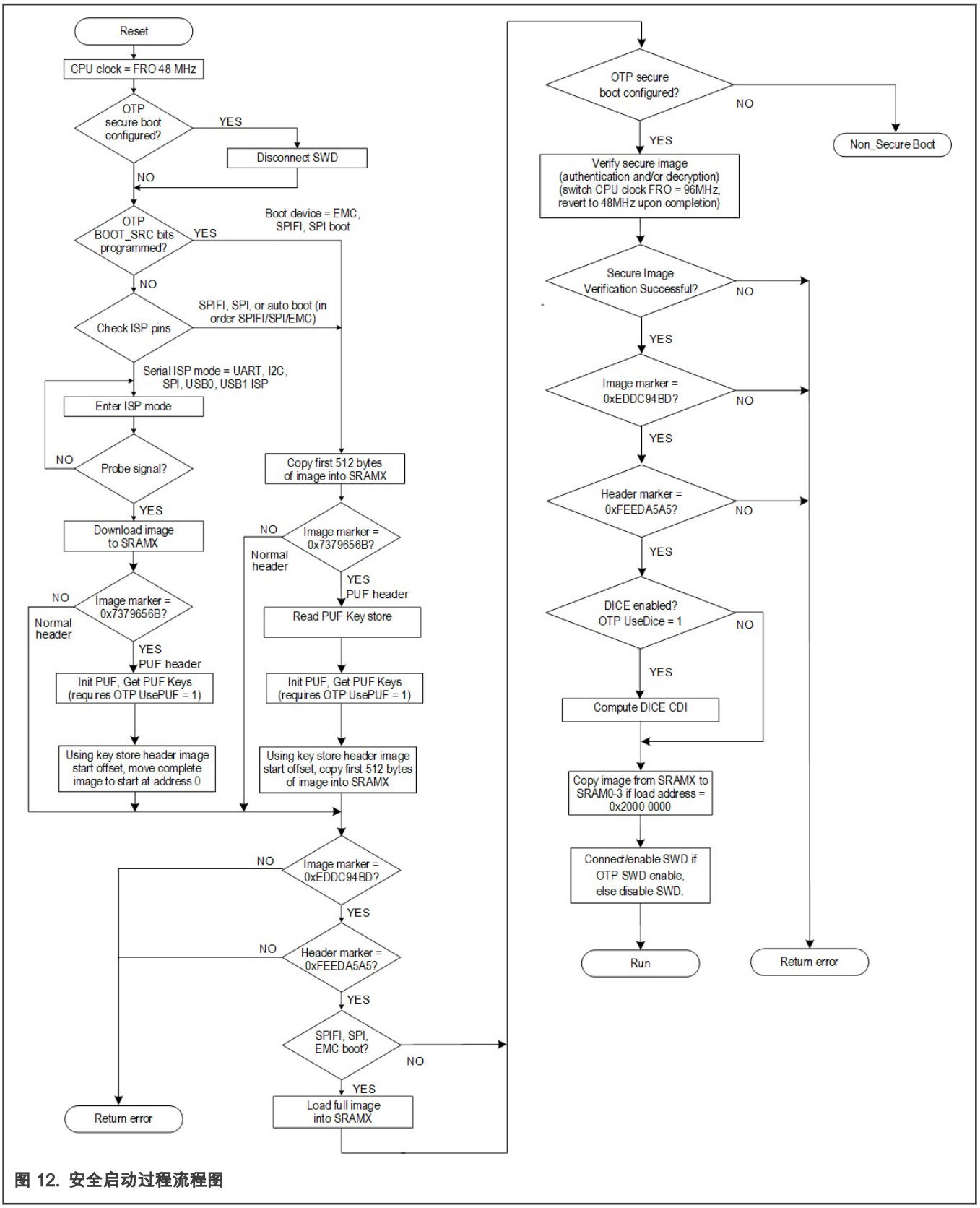

图 12. 安全启动过程流程图

3.3 启动密钥存储

密码密钥是密码算法用来将明文转换为密文或密文转换为明文的一串比特数据。该密钥保持私密性，并确保安全通信。 需要存储两种密钥：RoTKPublic 的 SHA256 和 AES 密钥。RoTKPublic 的 SHA256 是 RoT 公钥的 SHA-256 摘要，用于认证签名镜像。AES 密钥用于解密加密的镜像，如果存储在 OTP 存储区中则被加扰。

表 1 列出了带有相应镜像的密钥存储。

表 1. 密钥储存

密钥储存	签名镜像	加密镜像	签名加密镜像		加密签名镜像	
	SHA256 of RoTKPublic	AES Key	SHA256 of RoTKPublic	AES Key	SHA256 of RoTKPublic	AES Key
In OTP (OTP_USE_PUF = 0)	OTP Bank 1: Low 128-bit; OTP Bank 2: Upper 128-bit.	OTP Bank 2: 128-bit; scrambled AES Key[1]	OTP Bank 1: Lower 128-bit[2]	OTP Bank 2: 128-bit; scrambled AES Key[1]	OTP Bank 1: Lower 128-bit[2]	OTP Bank 2: 128-bit scrambled AES Key[1]
In PUF (OTP_USE_PUF = 1)	OTP Bank 1: Lower 128-bit; OTP Bank 2: Upper 128-bit[3].	PUF Key Store: 256-bit AES Key	OTP Bank 1: Lower 128-bit; OTP Bank 2: Upper 128-bit[3].	PUF Key store:256-bit AES key	OTP Bank 1: Lower 128-bit; OTP Bank 2: Upper 128-bit[3].	PUF Key store: 256-bit AES key

1. 软件无法读取加扰的 AES 密钥。
2. 仅将 SHA256 哈希的低 128 位与 OTP 存储区 1 内容进行比较，以验证 RoTKPublic。
3. 当将 OTP 存储区 2 用于以下条件时，仅将 SHA256 哈希的低 128 位与 OTP 存储区 1 的内容进行比较以验证 RoTKPublic，否则，将 256 位 SHA256 哈希与 OTP 存储区 1 和 2 进行比较。
   - OTP 存储区库 2 用于存储客户特定的 USB-IF 认证的供应商标识符（VID）和产品标识符（PID）。
   - 在 USB DFU 引导和 ISP 操作期间，引导 ROM 使用 USB ID。
   - 仅在最终产品中使用 USB DFU 引导时需要自定义 ID。
   - OTP 存储区 2 中 word0，应包含 USB ID 标记值 0x43555342。

3.4 设备标识符组合引擎（DICE）

LPC54S0xx 安全启动支持 Trusted Computing Group 规定的设备标识符组成引擎（DICE）规范，请参阅 Implicit Identity Based Device Attestation。

DICE 为每个 LPC54S0xx 器件提供了一种密码学方式生成 32 字节唯一 ID（称为复合设备标识符（CDI））的方法。DICE 在对用户镜像进行身份验证之后并将控制权转移到用户镜像之前计算 CDI。DICE 使用唯一设备密钥（UDS）和用户镜像的 SHA-256 摘要来计算 CDI。 CDI = HMAC（UDSKey，SHA2（SBL_IMG））， SBL_IMG = L0_IMAG 而没有 L0_Signature。UDS 来自 PUF 或唯一标识符（UUID）。对于基于 PUF 的 UDS，UDS 是使用密钥库中的密钥代码（在供应/制造过程中获得）检索的索引 15 密钥，并且 ROM 在 CDI 计算之后禁用 PUF 中的索引 15 密钥的解码。对于基于 UUID 的 UDS，将从 UUID 和 OTP 存储区 3（在 CDI 计算之后锁定寄存器 2/3）中计算出 UDS。退出安全启动并进入用户应用程序时，32 字节 CDI 保存在 SRAM0 上。

4 总结

LPC54S0xx 是一个安全的 MCU，集成了许多安全外围设备，例如 AES 引擎，SHA，RNG，PUF，DICE 等。安全启动驻留在 LPC54S0xx 的 ROM 中，以确保应用程序代码是真实的。所有这些高级安全功能使 LPC54S0xx 非常适合构建安全系统。

How To Reach Us

Home Page:

nxp.com

Web Support:

nxp.com/support

Limited warranty and liability — Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters that may be provided in NXP data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including “typicals,” must be validated for each customer application by customer's technical experts. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: nxp.com/SalesTermsandConditions.

Right to make changes - NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Security — Customer understands that all NXP products may be subject to unidentified or documented vulnerabilities. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer’s applications and products. Customer’s responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer’s applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately. Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP. NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

For more information, please visit: http://www.nxp.com

For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 2019 年 3 月 15 日

Document identifier: AN12385

arm