Trusty TEE

Trusty 是一种安全的操作系统 (OS)，可为 Android 提供可信执行环境 (TEE)。Trusty 操作系统与 Android 操作系统在同一处理器上运行，但 Trusty 通过硬件和软件与系统的其余组件隔离开来。Trusty 与 Android 彼此并行运行。Trusty 可以访问设备主处理器和内存的全部功能，但完全隔离。隔离可以保护 Trusty 免受用户安装的恶意应用以及可能在 Android 中发现的潜在漏洞的侵害。

Trusty 与 ARM 和 Intel 处理器兼容。在 ARM 系统中，Trusty 使用 ARM 的 Trustzone™ 虚拟化主处理器，并创建安全的可信执行环境。使用 Intel 虚拟化技术的 Intel x86 平台也提供类似的支持。

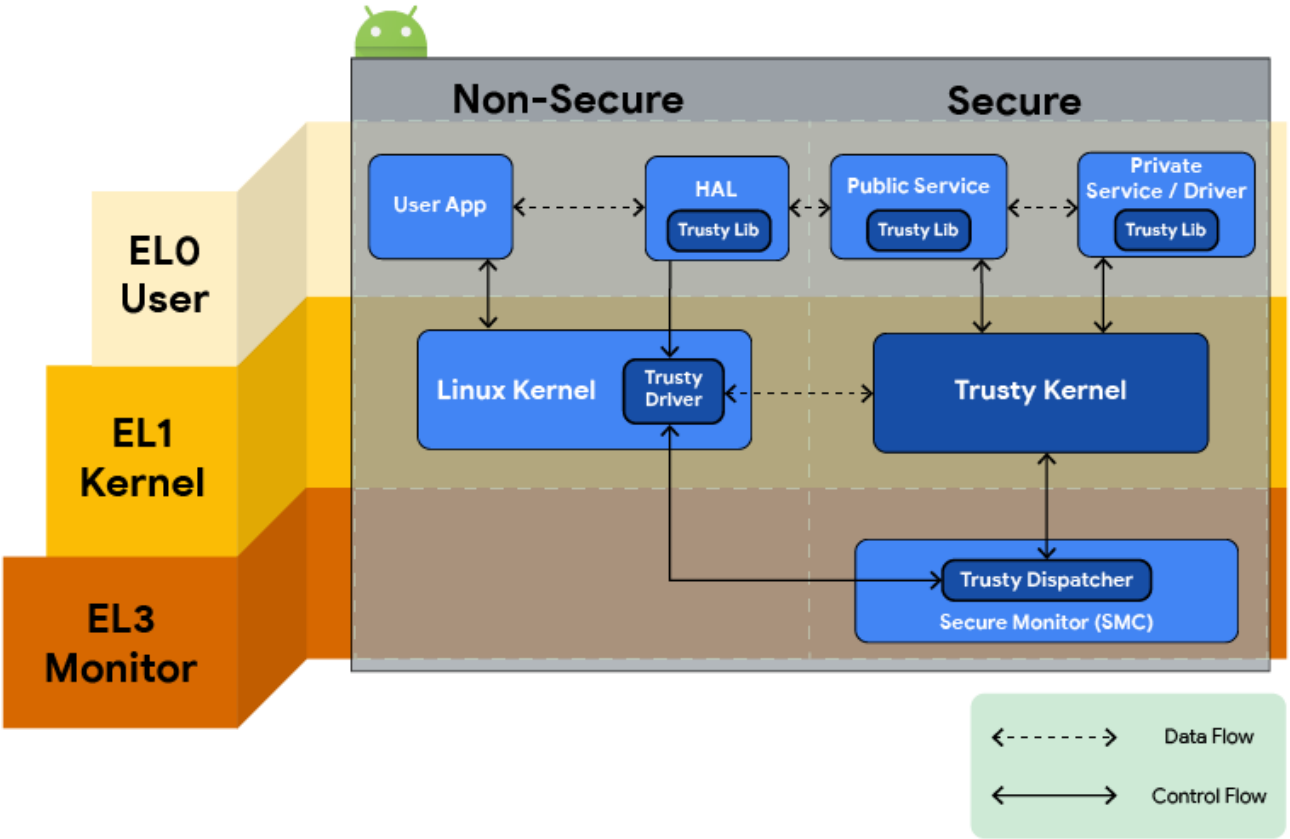

图 1. Trusty 概览图

Trusty 包含以下组件：

- 由 Little Kernel (https://github.com/littlekernel/lk) 衍生的小型操作系统内核
- Linux 内核驱动程序，用于在安全环境和 Android 之间传输数据
- Android 用户空间库 (https://android.googlesource.com/trusty/lib/)，用于通过内核驱动程序与可信应用（即安全任务/服务）通信

注意：Trusty 和 Trusty API 随时可能发生变化。如需了解 Trusty API，请参阅 API 参考（/security/trusty/trusty-ref）。

为什么使用 Trusty?

其他 TEE 操作系统历来都是由第三方供应商以二进制 Blob 的形式提供，或由内部开发。对系统芯片 (SoC) 供应商和 OEM 来说，开发内部 TEE 系统或从第三方获取 TEE 许可的成本可能很高。资金成本加上不可靠的第三方系统会导致 Android 生态系统不稳定。我们将 Trusty 作为一种可靠且免费的开源替代方案提供给合作伙伴，用于替代其可信执行环境。Trusty 可提供通过封闭源代码系统无法实现的透明性。

Android 支持各种 TEE 实现，因此您并非只能使用 Trusty。每一种 TEE 操作系统都会通过一种独特的方式部署可信应用。对试图确保应用能够在所有 Android 设备上正常运行的可信应用开发者来说，这种不统一性可能是一个问题。使用 Trusty 作为标准，可以帮助应用开发者轻松地创建和部署应用，而不用考虑有多个 TEE 系统的不统一性。借助 Trusty TEE，开发者和合作伙伴能够实现透明化、进行协作、检查代码并轻松地进行调试。可信应用的开发者可以集中利用各种常见的工具和 API，以降低引入安全漏洞的风险。这些开发者可以确信自己能够开发应用并让此应用在多个设备上重复使用，而不需要进一步进行开发。

应用和服务

Trusty 应用定义为二进制文件（可执行文件和资源文件）、二进制清单和加密签名的集合。在运行时，Trusty 应用在 Trusty 内核下以隔离进程的形式在非特权模式下运行。每个进程都会利用 TEE 处理器的内存管理单元功能在各自的虚拟内存沙盒中运行。硬件构建会改变 Trusty 遵循的确切过程，但是，举例来说，内核会使用由安全计时器驱动且按优先级进行调度的循环调度程序安排这些进程。所有 Trusty 应用均具有相同的优先级。

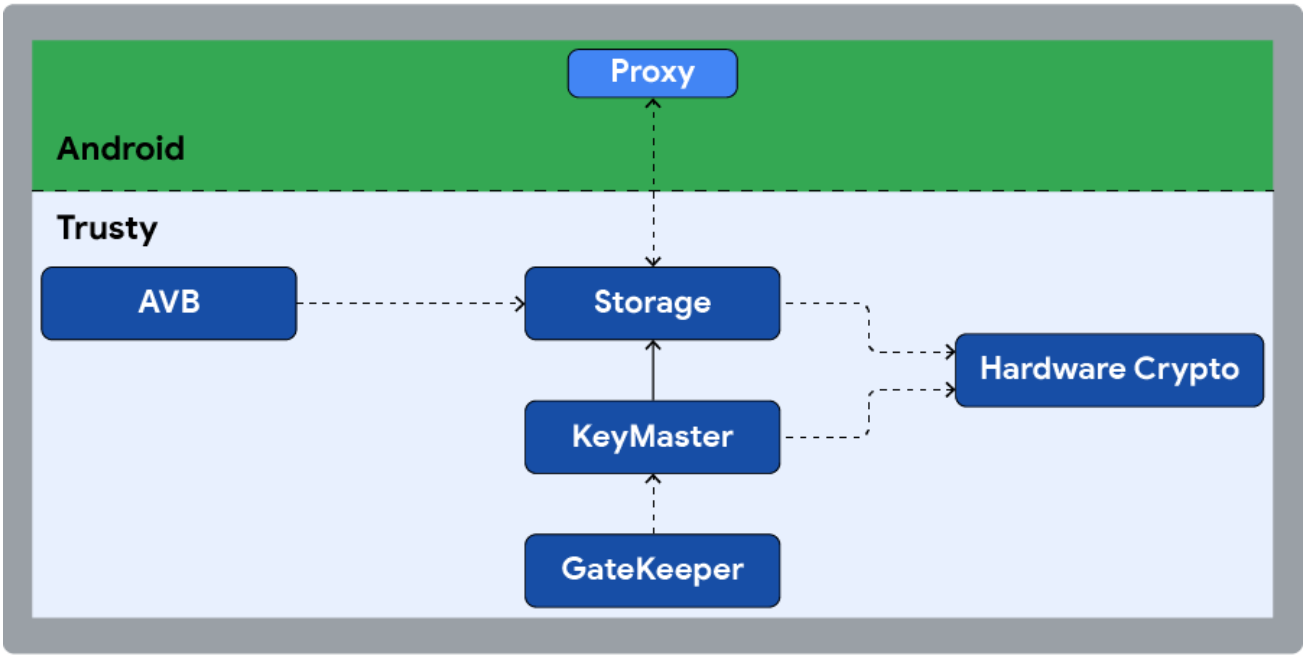

图 2. Trusty 应用概览。

第三方 Trusty 应用

目前，所有 Trusty 应用都是由一个开发方开发的，并用 Trusty 内核映像进行封装。整个映像都经过签名并在启动过程中由引导加载程序进行验证。目前，Trusty 不支持第三方应用开发。尽管 Trusty 支持开发新应用，但在开发新应用时务必要万分谨慎；每个新应用都会使系统可信计算基 (TCB) 的范围增大。可信应用可以访问设备机密数据，并且可以利用这些数据进行计算或数据转换。能够开发在 TEE 中运行的新应用为创新带来了多种可能性。不过，根据 TEE 的定义，如果这些应用没有附带某种形式的可信凭据，则无法分发。这种凭据通常采用数字签名的形式，即由应用运行时所在产品的用户信任的实体提供的数字签名。

用途和示例

可信执行环境正迅速成为移动设备领域的一项标准。用户的日常生活越来越依赖移动设备，安全需求也在不断增长。具有 TEE 的移动设备比没有 TEE 的设备更安全。

在具有 TEE 实现的设备上，主处理器通常称为“不可信”处理器，这意味着它无法访问制造商用于存储机密数据（例如设备专用加密秘钥）的特定 RAM、硬件寄存器和一次写入 Fuse 区域。在主处理器上运行的软件会将所有需要使用机密数据的操作委派给 TEE 处理器。

在 Android 生态系统中，最广为人知的示例是受保护内容的 DRM 框架 (/devices/drm.html)。在 TEE 处理器上运行的软件可以访问解密受保护内容所需的设备专用密钥。主处理器只能看到已加密的内容，这样一来，就可以针对软件类攻击提供高级别的安全保障和保护。

TEE 还有许多其他用途，例如移动支付、安全银行、多重身份验证、设备重置保护、抗重放攻击的持久存储、安全 PIN 和指纹处理，甚至还能用于恶意软件检测。