瑞萨 RA 产品家族

器件生命周期管理密钥安装

简介

器件生命周期管理 (DLM) 系统能够对产品从开发伊始到生产再到使用寿命结束的整个过程进行管理。RA 产品家族 MCU 调试功能和串行编程功能由器件生命周期状态决定。

第一代 RA DLM 系统使用明文 MCU 调试识别码 (ID)，适用于不支持 TrustZone® 的 RA 产品家族 MCU 。新一代 DLM 系统利用需要经过身份验证且基于封装密钥管理的器件生命周期状态转换过程，适用于基于 Arm® TrustZone® 的 RA 产品家族 MCU。这种新一代 DLM 系统可提供增强的 IP 保护功能，同时还具有使器件生命周期状态回退的功能。

本应用笔记重点介绍了基于新一代 TrustZone® 的 RA 产品家族 MCU DLM 系统的用例，将会引导您了解 DLM 密钥封装服务、密钥安装步骤和器件生命周期回退步骤。此外，还会简要介绍第一代 DLM 系统。

本文以 RA6M4 为例，对密钥封装、密钥安装和生命周期状态回退进行讲解。文中所述的一般步骤适用于所有基于 TrustZone® 的 RA 产品家族 MCU。

所需资源

本应用笔记参考了以下资源：

开发工具和软件

- 瑞萨闪存编程器 (RFP) 版本 3.08 或更高版本
  https://www.renesas.com/us/en/products/software-tools/tools/programmer/renesas-flash-programmer-programming-gui.html
- e² studio IDE 版本 2020 10 或更高版本
- RA 产品家族灵活配置软件包 (FSP) 版本 2.0.0 或更高版本
- SEGGER J-Link® USB 驱动程序版本 6.86 或更高版本

FSP、J-Link USB 驱动程序和 e² Studio 以捆绑包的形式通过一个可下载的平台安装程序提供，可从 FSP 网页（网址为 renesas.com/ra/fsp）上下载。

硬件

- EK-RA6M4，用于 RA6M4 MCU 系列的评估板 (renesas.com/ra/ek-ra6m4)
- 运行 Windows® 10 操作系统的测试用 PC
- 一根 USB 线缆（Type-A 公头转 micro-B 公头）

前提条件和目标受众

本应用笔记假设您在使用瑞萨的 e² studio IDE 和闪存编程器 (RFP) 方面有一定的经验。此外，本应用笔记假设您对 RA 产品家族 MCU 的安全功能有一定的了解。有关背景信息，请参见*《瑞萨 RA6M4 MCU 系列用户手册：硬件》*的安全功能部分。

目标受众包括产品开发人员、产品制造商、产品支持人员或参与 RA 产品家族 MCU 器件生命周期管理任何阶段的最终用户。

目录

1. RA 产品家族 MCU 系列的器件生命周期管理简介

RA 产品家族 DLM 系统可以在客户应用程序开发、生产、产品部署和故障分析管理中发挥关键作用。

1.1 使用调试 ID 码的器件生命周期管理

在器件部署完成后，第一代 RA 产品家族 MCU 系列使用调试识别码（ID 码）来重新使能调试接口。产品开发通常由一个受信任的软件团队管理。

下图所示为基于 Arm® Cortex®-M0+ 和 M4 的 RA 产品家族 MCU 系列的典型生命周期管理。

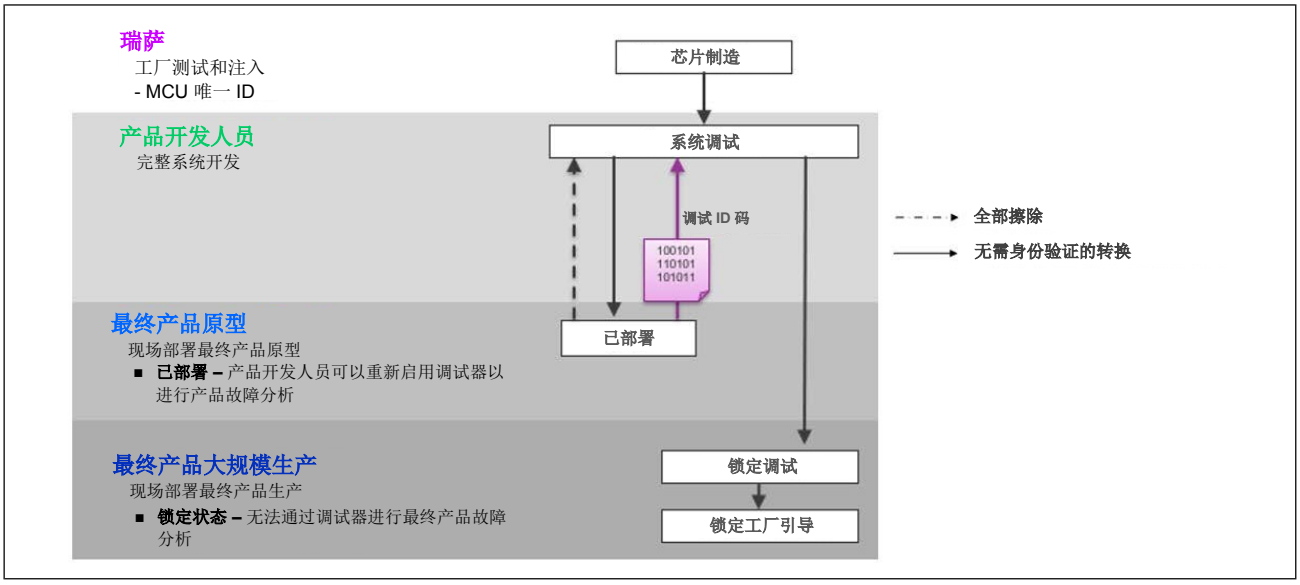

图 1. RA 产品家族 MCU 系列的 DLM 系统（使用调试 ID 码）

- 从瑞萨工厂收到器件后，产品开发人员可以创建对闪存、RAM 和外设等 MCU 资源具有完全访问权限的应用程序代码。
- 开发完成后，产品开发人员会为产品原型设置 ID 码以进行临时保护。这可以禁止任何在调试器连接时没有提供 ID 码的连接尝试，防止进一步访问调试系统。如果要进行故障分析，最终产品用户可以将 MCU 发回给产品开发人员。产品开发人员可以使用 ID 码重新启用调试器以进行故障分析。
- 要进行大规模生产，制造商可以对产品进行编程，并设置 ID 码以永久禁止访问调试系统。工厂引导加载程序也可以通过闪存访问窗口 (FAW) 和安全 MPU 永久禁用。有关 ID 码保护、FAW 设置和安全 MPU 设置的操作步骤，请参见*使用安全 MPU 保护静态数据*应用程序项目示例。

这种 DLM 系统可以为许多应用程序提供足够的保护，但由于解锁ID码是以纯文本形式传输的，因此容易遭到窃听和重放攻击。

1.2 使用 Arm® TrustZone® 技术的器件生命周期管理系统

新一代 RA 产品家族 MCU 具有增强的 DLM 功能。基于 TrustZone 技术的第一个增强功能是能够将应用程序分成两个区域，通常称为安全/受信任区域和非安全/非受信任区域。这种分区的一个典型用例是隔离产品的信任根，包括器件的身份和可以访问或更改器件身份的支持服务。在开发完安全系统并将其烧录到器件中后，生命周期状态从安全系统调试 (SSD) 转为非安全系统调试 (NSECSD)。这允许在保护信任根本身的同时，开发和调试使用信任根提供的功能的应用程序。

1.2.1 器件生命周期状态定义

器件生命周期确定了器件的当前所处的阶段，并控制调试接口、串行编程接口和瑞萨测试模式的访问权限。新一代 RA 产品家族 MCU 可提供加密身份验证以恢复各种级别的编程和调试功能。在生产过程中，MCU 会配备一个或多个 DLM 状态密钥。当开发人员申请 DLM 状态回退以解锁 MCU 时，MCU 会发出质询。然后，开发人员必须使用适当的 DLM 状态密钥来构建响应。如果响应正确，MCU 将退回到申请的 DLM 状态，同时解锁定义的编程和调试功能。这种机制可以防止窃听和重放攻击，提供增强的 IP 保护，同时保留最终产品执行故障分析的功能。

基于 TrustZone® 的 RA 产品家族 MCU 有三个调试级别：

- DBG2：允许调试器连接，可以无限制访问存储器和外设
- DBG1：允许调试器连接，但仅限访问非安全内存区域和外设
- DBG0：不允许调试器连接

当器件处于引导模式时，串行编程接口可以与器件的工厂引导加载程序通信。用户可以使用瑞萨闪存编程 (RFP) 应用程序通过串行编程接口与工厂引导加载程序通信，以更新器件生命周期状态。

下表列出了各个器件生命周期状态的定义和相应的调试级别，以及串行编程接口的功能。

表 1. 基于 TrustZone 的 RA 产品家族 MCU 系列器件生命周期状态

生命周期状态	定义和状态特性	调试级别	串行编程	瑞萨测试模式
CM	• “**C**hip **M**anufacturing” “芯片制造” • 器件在瑞萨工厂中。 • 开发人员收到处于该状态的器件。 • 已经写入 MCU 唯一 ID 和硬件唯一密钥 (HUK)。	DBG2	• 可用。 • 无法访问代码/数据闪存区域。	不可用
SSD	• “**S**ecure **S**oftware **D**evelopment” “安全软件开发” • 应用程序的安全部分正在开发中。 • 可以注入 SECDBG_KEY 和 RMA_KEY。	DBG2	• 可用。 • 可以编程/擦除/读取全部代码/数据闪存区域。	不可用
NSECSD	• “**N**on-**SEC**ure **S**oftware **D**evelopment” “非安全软件开发” • 应用程序的非安全部分正在开发中。 • 可以注入 NONSECDBG_KEY。 • 如果在 SSD 状态下注入了 SECDBG_KEY，则可以在不擦除闪存的情况下退回到 SSD 状态。 • 可以通过完全擦除闪存以退回到 SSD 状态。	DBG1	• 可用。 • 可以编程/擦除/读取非安全代码/数据闪存区域。	不可用

生命周期状态	定义和状态特性	调试级别	串行编程	瑞萨测试模式
DPL	• “**DePL**oyed” “已部署” • 器件已经投放市场。 • 如果在 NSECSD 状态下注入了 NONSECDBG_KEY，则可以在不擦除闪存的情况下退回到 NSECSD 状态。 • 可以通过完全擦除闪存以退回到 SSD 状态。	DBG0	• 可用。 • 无法访问代码/数据闪存区域。	不可用
LCK_DBG	• “**LoCK**ed **DeBuG**” “锁定调试” • **重要：调试接口已永久禁用。**	DBG0	• 可用 • 无法访问代码/数据闪存区域。	不可用
LCK_BOOT	• “**LoCK**ed **BOOT** interface” “锁定引导接口” • **重要：调试接口和串行编程接口已永久禁用。**	DBG0	• 不可用。	不可用
RMA_REQ	• “**R**eturn **M**aterial **A**uthorization **REQ**uest” “申请退货授权” • 申请退货授权 (RMA)。 • 在该状态下，客户必须将器件发送给瑞萨。	DBG0	• 可用。 • 无法访问代码/数据闪存区域。	不可用
RMA_ACK	• “**R**eturn **M**aterial **A**uthorization **ACK**nowledged” “退货授权已确认” • 在瑞萨进行故障分析。	DBG2	• 可用。 • 无法访问代码/数据闪存区域。	可用

1.2.2 器件生命周期状态和转换摘要

图 2 汇总了 MCU 的整个生命周期中所有可能的状态转换。

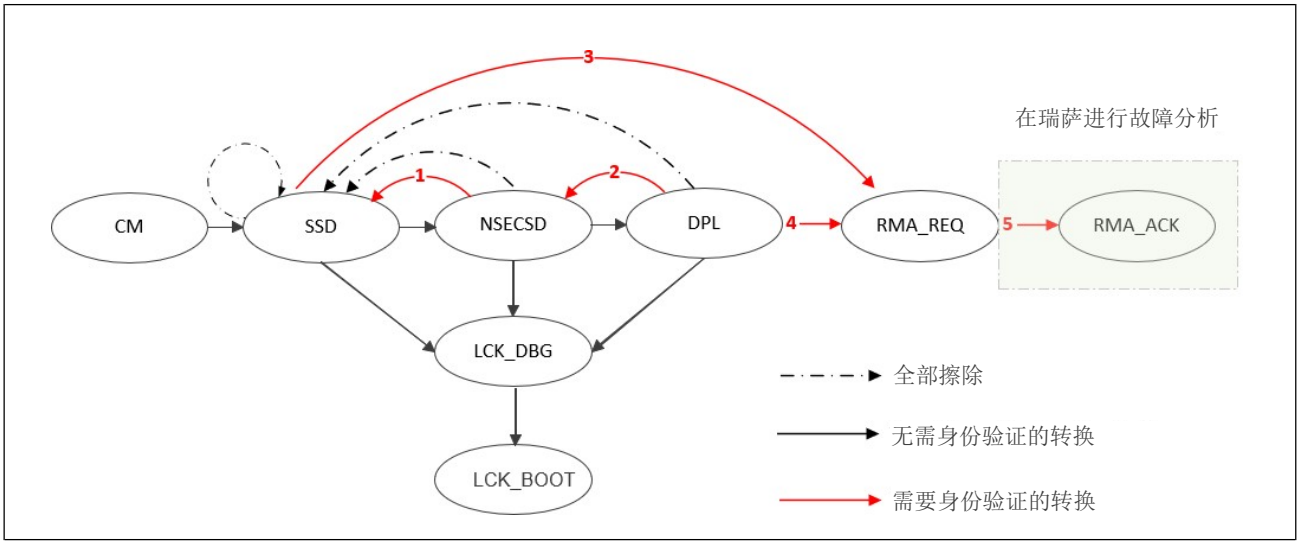

图 2. RA MCU 器件生命周期

如图 2 所示，共有三种类型的转换。

“All Erase”（全部擦除）操作

- 成功执行“**All erase**”（全部擦除）命令会将器件生命周期状态更改回 SSD。
- 如果闪存块锁定是临时的而非永久的，则“**All erase**”（全部擦除）命令将擦除整个闪存。
- 如果开发人员已强制执行永久块保护，则不会执行“**Initialize**”（初始化）命令。
- 可以通过 RFP 中的“**Initialize**”（初始化）命令执行“**All erase**”（全部擦除），除非“**Initialize**”（初始化）命令本身已被禁用。任何人均可使用“**Initialize**”（初始化）命令，因此闪存中的内容很容易被擦除。
- 如果开发人员不想使用此功能，可以使用 RFP 永久禁用“**Initialize**”（初始化）命令，如图 8 所示。但是，一旦禁用“**Initialize**”（初始化）命令，则永远无法恢复。
- 制造商可以在生产期间根据需要禁用“**Initialize**”（初始化）命令。
- 有关操作详情，请参见第 1.2.4 节和第 1.2.5 节。

DLM 身份验证密钥

如图 2 所示，需要 DLM 密钥来使生命周期状态回退，无需擦除 MCU 闪存的内容即可进行故障分析。DLM 密钥包括 RMA_KEY、SCEDBG_KEY 和 NONSCEDBG_KEY。

- 如表 1 中所述，安全开发人员可以将 SECDBG_KEY 和 RMA_KEY 注入到 SSD 状态下的器件中。RMA_KEY 可用于最终产品 RMA，SECDBG_KEY 可用于使器件返回到安全开发状态。
- 如表 1 中所述，非安全开发团队可以在 MCU 处于 NSECSD 状态时将 NONSECDBG_KEY 注入其中。NONSECDBG_KEY 可用于使器件返回到非安全开发状态。

需要经过身份验证的转换

需要经过身份验证的转换通常用于开发期间和部署之后的故障分析。故障分析通常需要使器件生命周期状态退回到之前的开发状态或前进到申请退货授权状态。

- 以下生命周期状态转换可以退回到之前的开发状态，无需擦除闪存内容即可进行故障分析：
  - 安全开发 (SSD) 转换到非安全开发 (NSECSD)
    - 转换 1：使用 SECDBG_KEY
  - 原型已部署 (DPL) 状态转换到非安全开发 (NSECSD)
    - 转换 2：使用 NONSECDBG_KEY
- 此外，可以从原型已部署状态或安全开发状态前进到申请退货授权状态，此时需要擦除闪存上的内容，闪存块已永久锁定的情况除外。
  - 原型已部署 (DPL) 状态转换到申请退货授权 (RMA_REQ)
    - 转换 3：使用 RMA_KEY
  - 安全开发 (SSD) 状态转换到申请退货授权 (RMA_REQ)
    - 转换 4：使用 RMA_KEY 或 MCU 唯一 ID
- 转换 5 需要执行瑞萨的专有操作。本应用笔记中不提供有关此转换的详细信息。如需相关信息，请联系瑞萨销售代表。

有关器件生命周期转换用例的更多详细信息，请参见第 2 章。

1.2.3 基于 Arm® TrustZone® 的器件生命周期管理的优势

以下是基于 TrustZone 的器件生命周期管理系统的摘要：

- 保护信任根和 IP 系统
- 无需擦除 MCU 闪存即可重新使能调试接口和串行编程接口以启用故障分析
- 擦除整个闪存以允许器件返回到安全软件开发阶段，以免废弃 MCU
- 防止窃听和重放攻击，提供增强的 IP 保护，同时保留执行最终产品故障分析的能力。

1.2.4 使用瑞萨器件分区管理器进行器件生命周期状态管理概述

瑞萨器件分区管理器是一个与 e² studio 集成的实用程序，用于在生产开发期间进行器件生命周期状态管理。用户可以使用瑞萨器件分区管理来执行以下功能：

- 查询当前器件生命周期
- 查询器件 IDAU 区域设置
- 将器件初始化以返回到 SSD 状态，擦除所有未锁定的闪存块
- 设置 IDAU 区域

请注意，如果使用 J-Link 作为连接接口，则在调试会话之后，用户需要对电路板重新上电，然后才能使用**瑞萨器件分区管理器**。

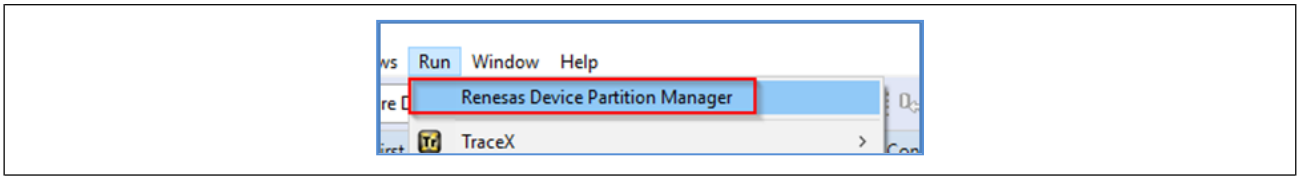

图 3. 打开瑞萨器件分区管理器

以下是执行“**Initialize device back to factory default**”（将器件初始化以返回到出厂默认设置）的设置示例。选择连接方式，然后单击“**Run**”（运行）。

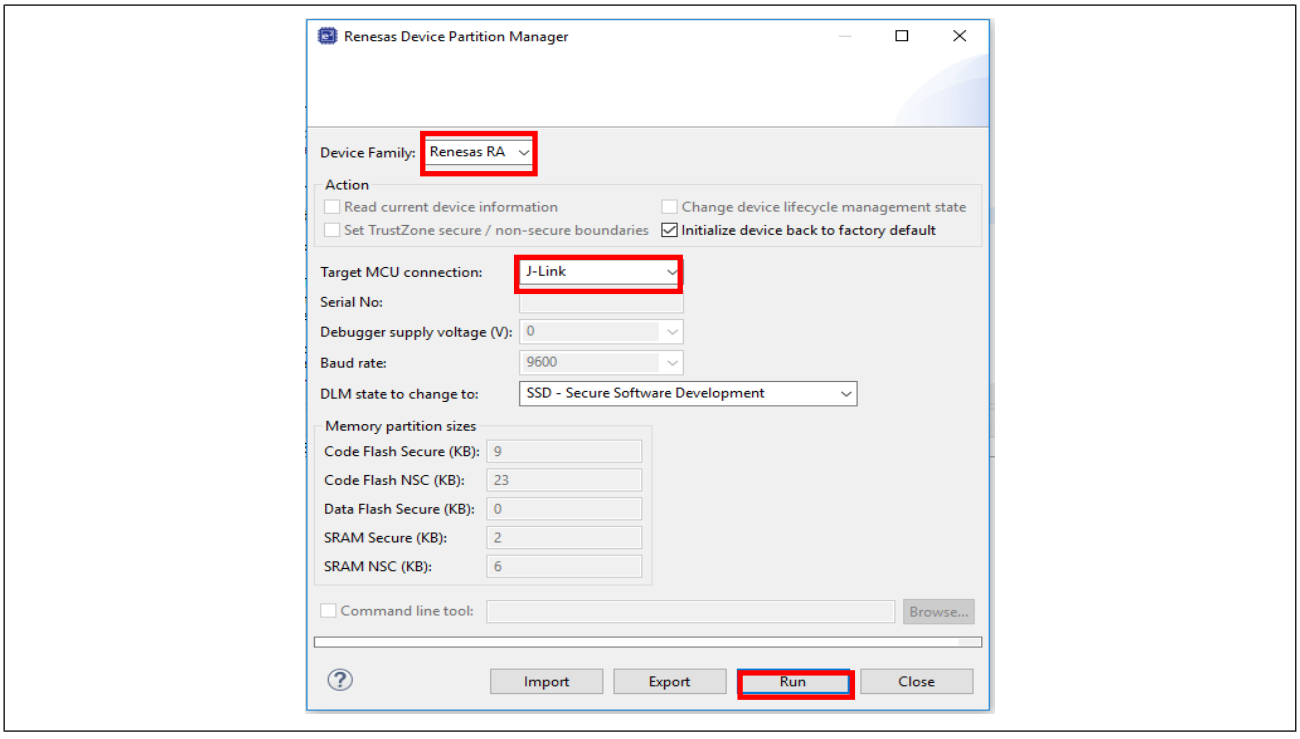

图 4. 使用瑞萨器件分区管理器初始化 RA6M4

在产品开发期间，用户通常使用**瑞萨器件分区管理器**推进器件生命周期：

- SSD -> NSECSD
- NSECSD-> DPL

瑞萨器件分区管理器的限制

- 瑞萨器件分区管理器不支持需要经过身份验证的转换。请使用 RFP 进行故障分析。
- 瑞萨器件分区管理器支持转换到有限的器件生命周期状态，并需要 e² studio IDE 环境来操作。有些大规模生产需要转换到 LCK_DBG 或 LCK_BOOT，在这种情况下，用户应使用 RFP。

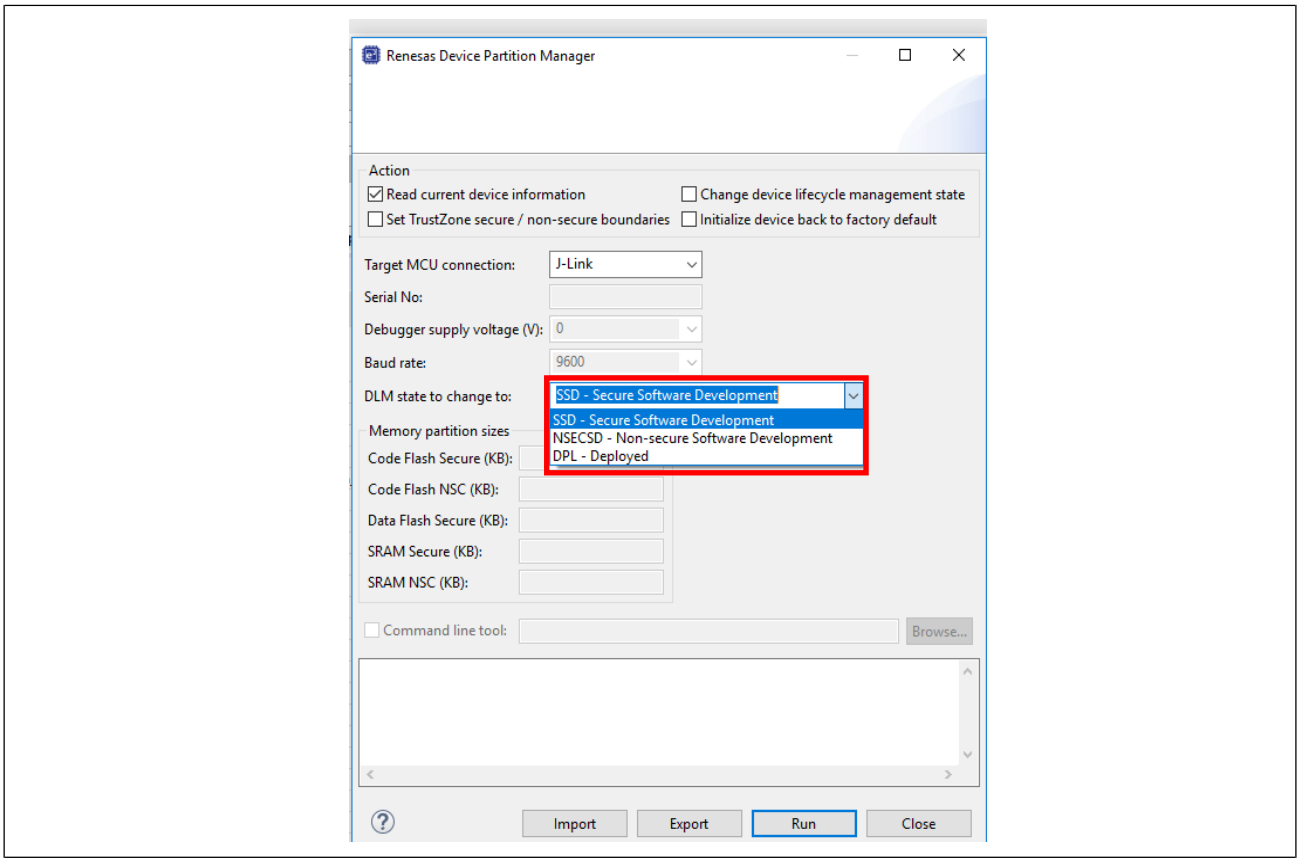

图 5. 使用瑞萨器件分区管理器推进器件生命周期状态

1.2.5 使用瑞萨闪存编程器的器件生命周期状态转换概述

瑞萨闪存编程器可提供端到端的生产流程支持。除了瑞萨器件分区管理器提供的器件生命周期管理功能外，RFP 还提供以下功能。

- 支持需要经过身份验证的器件生命周期状态转换。需要在身份验证过程中安装和使用 DLM 密钥。有关使用 RFP 进行需要经过身份验证的转换的信息，请参见第 3.6 节。
- 支持所有不进行身份验证的器件生命周期状态转换，转换到 RMA_ACK 除外（该转换只能在瑞萨内部由授权团队执行，如图 2 所示）。
- 禁用“**Initialize**”（初始化）命令。如果器件在 DPL 状态下部署，并且需要避免意外擦除闪存内容，则可能需要执行上述操作。但是，一旦禁用“**Initialize**”（初始化）命令，则永远无法恢复。

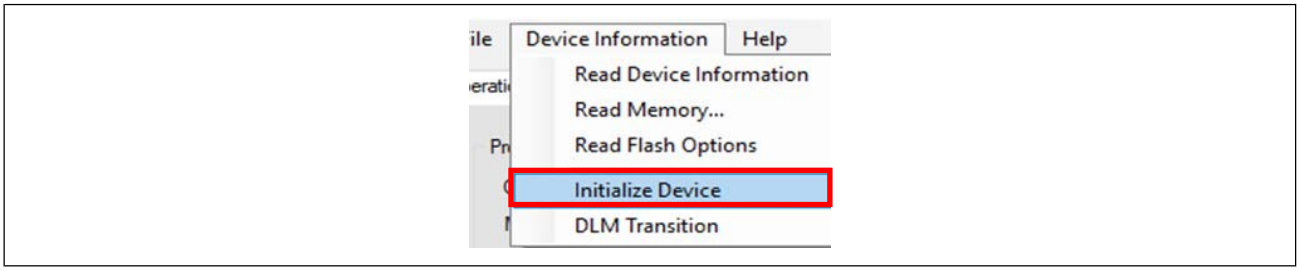

图 6. 使用“Initialize Device”（初始化器件）命令执行“All erase”（全部擦除）

对于不进行身份验证的转换，请使用“**DLM Transitions**”（DLM 转换）来执行转换，如图 7 所示。

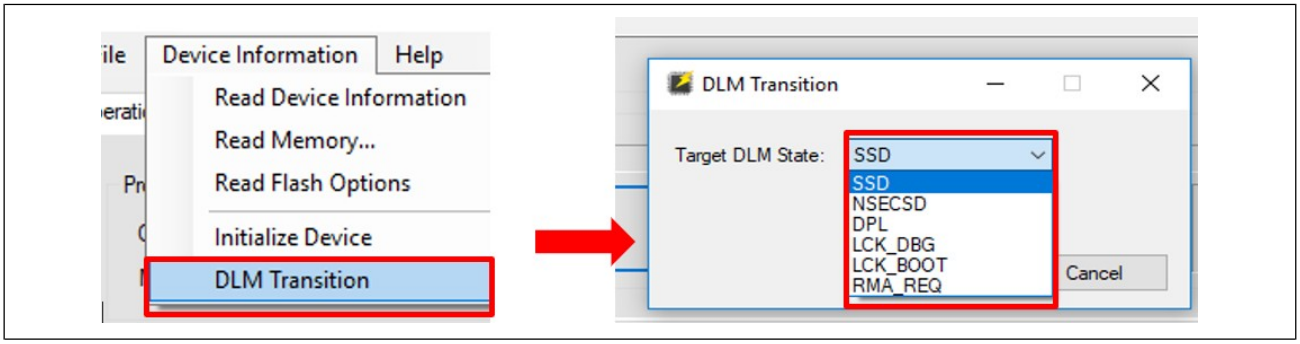

图 7. RFP 支持的可用生命周期状态转换

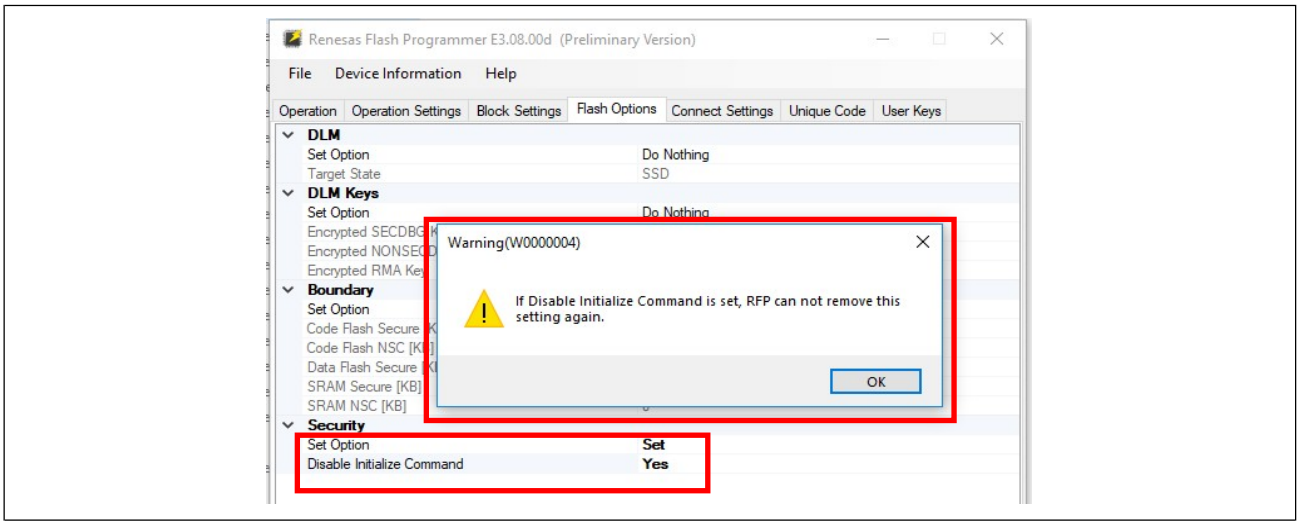

图 8. 禁用“Initialize”（初始化）命令

2. 基于 Arm® TrustZone® 的应用程序中的器件生命周期管理系统用例

本章将介绍基于 TrustZone 技术的瑞萨产品的器件生命周期管理系统用例，涵盖了产品的开发、生产和部署阶段。

瑞萨 RA IDE 支持三种开发模型，均用于基于 TrustZone 技术的 MCU。瑞萨 RA 项目生成器提供三种项目类型来支持这三种开发模型，如图 9 所示。

- 扁平化项目开发模型
  - 开发过程中没有内置对 TrustZone 技术的感知功能
  - 在生产阶段，器件生命周期管理可带来诸多优势
- 分离式项目开发模型
  - 使用安全和非安全项目类型
  - 这种开发模型允许由两个团队进行产品开发。

  - 安全开发人员 - 创建信任根 (5.1) 或孤立的子系统。开发是在 SSD 状态下进行的。
  - 非安全开发人员 - 创建基于信任根构建并使用孤立子系统的应用程序。开发是在 NSECSD 状态下进行的。
  - 非安全开发人员无法访问安全系统。
- 联合式项目开发模型
  - 使用安全和非安全项目类型。
  - 安全和非安全应用程序开发均由一个受信任团队在 SSD 状态下进行。
  - 开发人员可以访问安全和非安全资源（硬件、代码、数据和调试）。

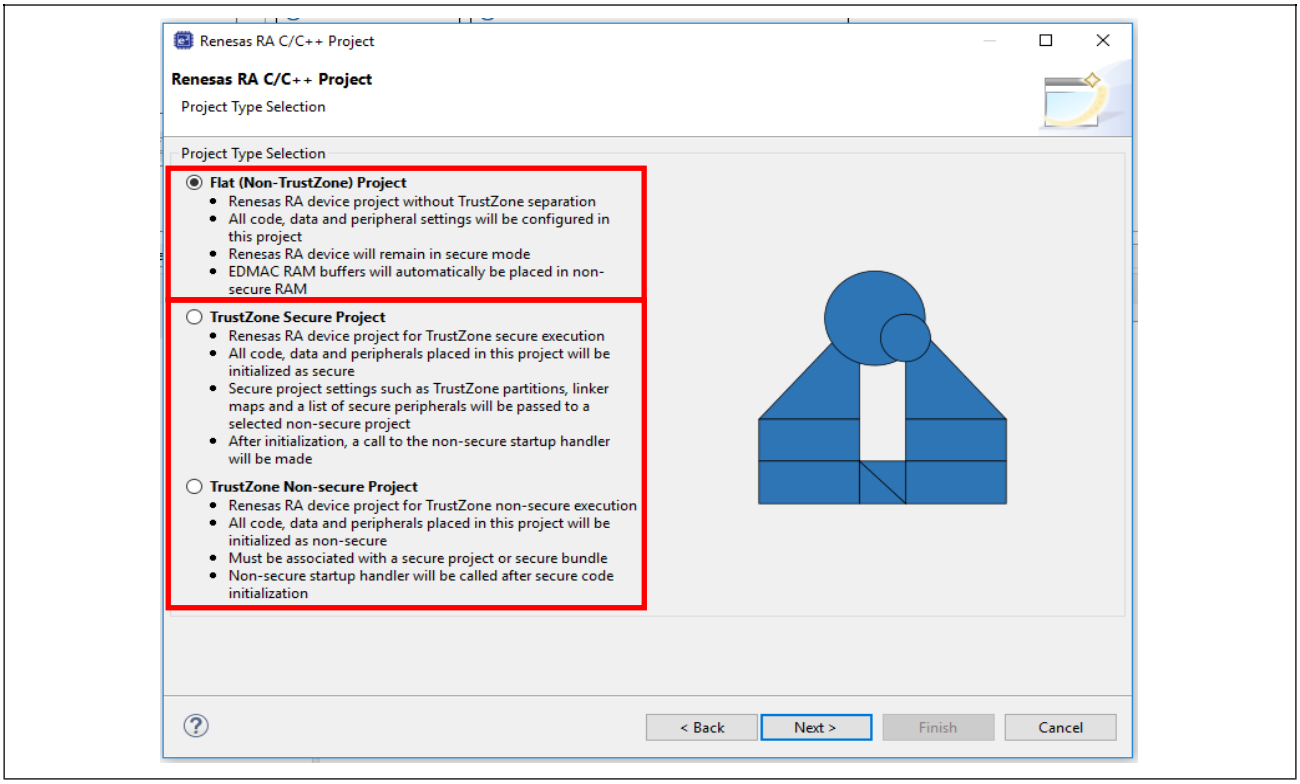

图 9. 基于 Arm® TrustZone® 技术的瑞萨 RA 项目类型

2.1 基于 TrustZone 技术的开发模型概述

使用基于 TrustZone 技术的 RA 产品家族 MCU 系列进行开发时，通常使用前文所述的两种开发模型：分离式项目开发模型和联合式项目开发模型。本节将介绍这两种开发模型的器件生命周期开发用例。

2.1.1 分离式项目开发模型

分离式项目开发模型的一般流程如下所述：

- 在 MCU 制造的最后几个步骤中，瑞萨将在其中一个步骤向 MCU 注入 MCU 唯一 ID 和 HUK，然后将 MCU 交付给安全开发人员。
- 安全产品开发人员将开发安全应用程序，并锁定安全区域以防止非安全区域访问，然后将 MCU 交付给非安全开发人员。
- 非安全产品开发人员将开发非安全应用程序。在此阶段，非安全产品开发人员无法看到安全系统。
- 最终产品用户可能会收到调试接口临时锁定或永久锁定的 MCU。串行编程接口可能被永久锁定，或者可用但功能有限。

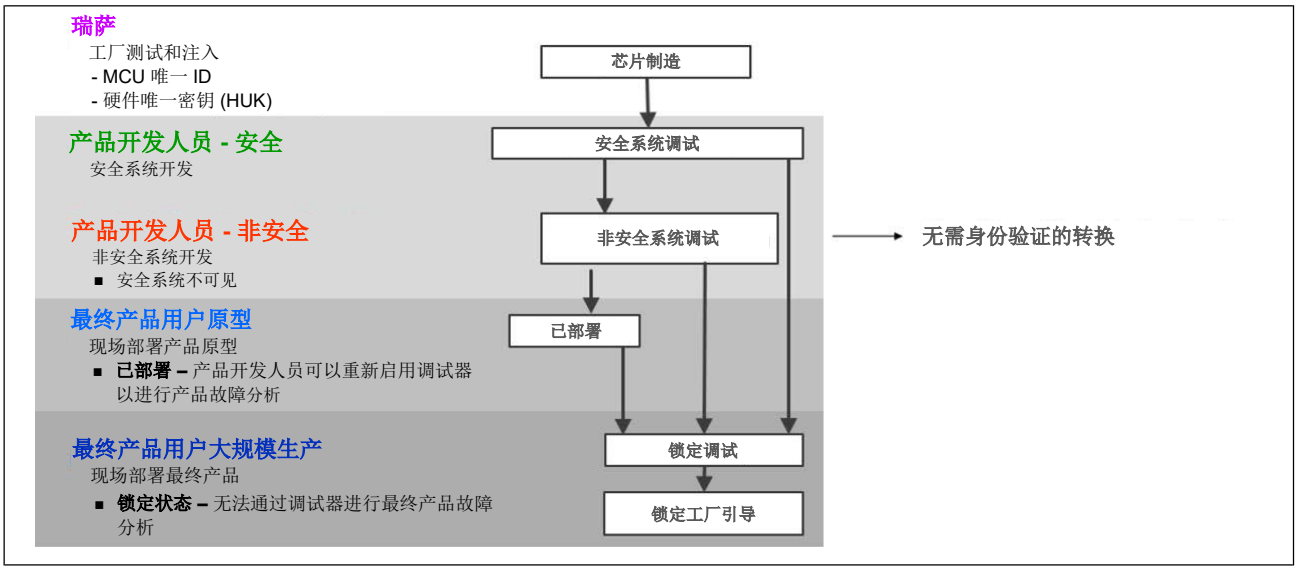

图 10. 分离式项目开发模型

开发、生产和故障分析期间的器件生命周期状态转换摘要

图 11 将生命周期状态添加到开发流程中，以全面了解分离式项目开发模型的开发、生产流程和故障分析过程中所有可能发生的转换。

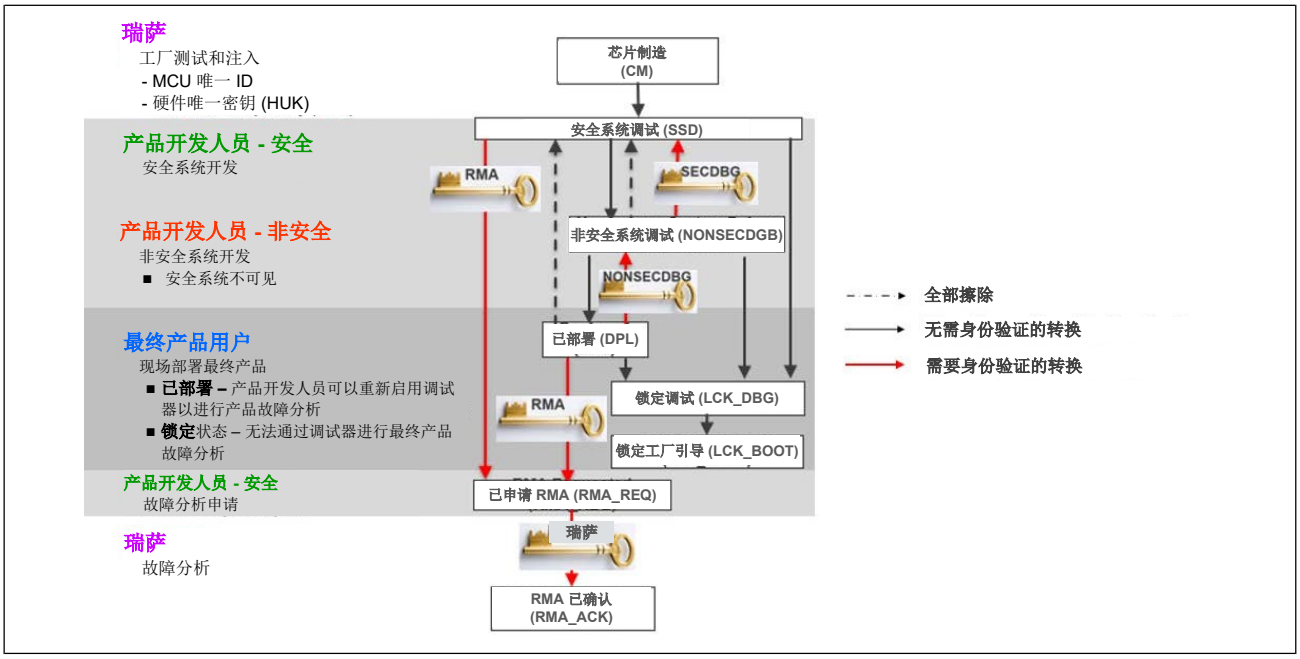

图 11. 分离式项目开发模型中的器件生命周期状态

2.1.2 联合式项目开发模型

若使用联合式项目开发模型，安全和非安全应用程序开发均由一个受信任的团队负责。在联合式项目开发模型中，器件生命周期状态的前进和回退类似于分离式项目开发模型，主要区别如下所示：

- 产品开发人员将在 SSD 状态下开发安全和非安全系统。
- 产品开发人员可以访问整个 MCU 资源，其中包括安全和非安全资源。

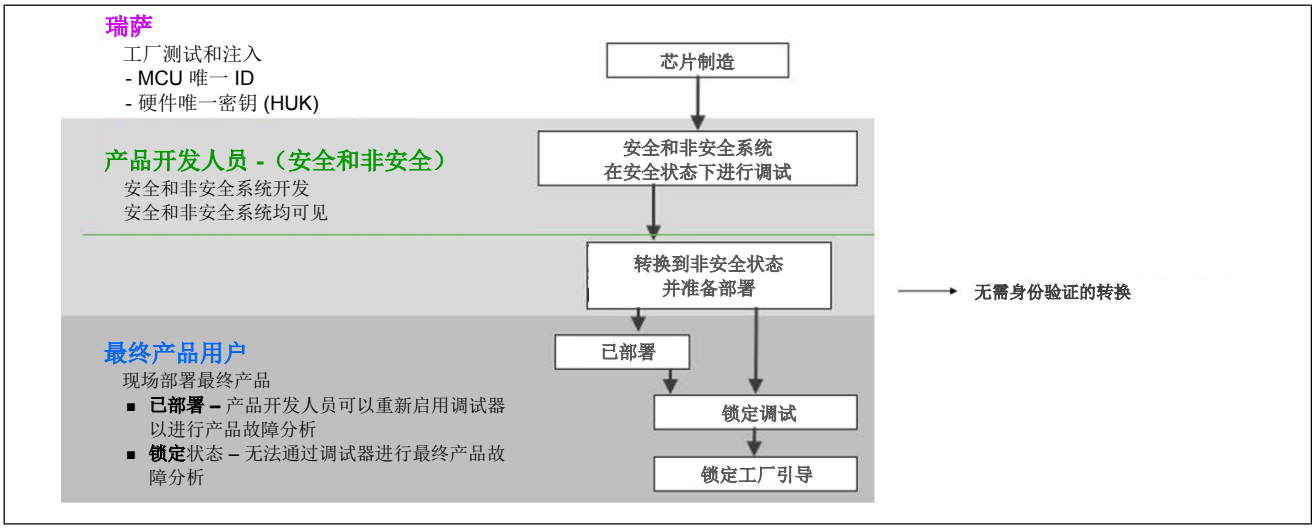

图 12. 联合式项目开发模型

开发、生产和故障分析期间的器件生命周期状态转换摘要

图 13 介绍了联合式项目开发模型的开发、生产流程和故障分析过程中所有可能发生的转换。

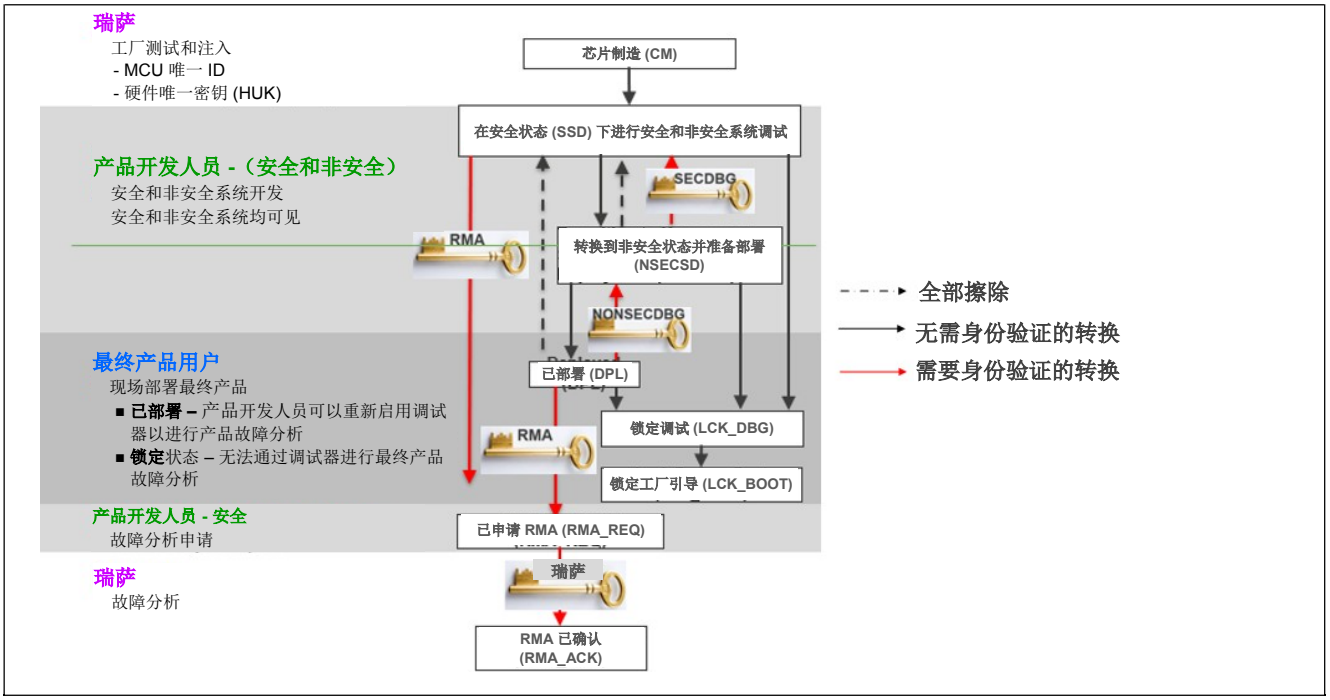

图 13. 联合项目开发模型中的器件生命周期状态

2.2 安全应用程序开发阶段的器件生命周期状态转换

在以下分析中，以分离式项目开发模型为例。每当分离式项目开发模型和联合式项目开发模型之间存在差异时，均会明确说明。

如图 11 所示，在 MCU 制造的最后几个步骤中，瑞萨将在其中一个步骤向 MCU 注入 MCU 唯一 ID 和 HUK，然后在 CM 状态下将 MCU 交付给安全开发团队。

如图 10 所示，可能的器件生命周期状态转换包括：

- 在开始安全应用程序开发之前：

  CM -> SSD：无需身份验证，无需擦除闪存。
- 在开发过程中，安全开发人员可以执行“**`All erase`**”（全部擦除）命令来恢复 MCU。生成的器件生命周期状态仍然是 SSD。

  SSD -> SSD：无需身份验证，将擦除整个闪存，但不会擦除永久锁定的闪存块。

  请注意，在 SSD 状态下，安全开发人员可以使用 RFP 来禁用“**`All erase`**”（全部擦除）命令。
- 在处于 SSD 生命周期状态期间注入 RMA_KEY 后，安全开发人员可以在成功完成质询响应身份验证后，使用 RMA_KEY 将 MCU 的生命周期更改为 RMA_REQ 状态。**SSD -> RMA_REQ**：使用 RMA_KEY 进行身份验证，将擦除整个闪存，但不会擦除永久锁定的闪存块。
- 在开发和测试安全应用程序之后：

  SSD -> NSECSD：无需身份验证，无需擦除闪存

  如果开发采用联合式项目开发模型，产品开发人员将在 SSD 状态下完成安全和非安全应用程序开发。

器件生命周期状态从 NSECSD 退回到 SSD（需要身份验证）

若之前已在 SSD 状态下安装 SECDBG_KEY，非安全开发人员可以将 MCU 返回给安全开发人员进行故障分析：

- 安全开发人员可以使用 SECDBG_KEY 将 MCU 状态从 NSECSD 更改为 SSD，并进行相应的调试或故障分析。
- 这种状态转换是需要经过验证的转换，在该状态更改中不会擦除闪存。**NSECSD -> SSD**：使用 SECDBG_KEY 进行身份验证，无需擦除闪存。

2.3 非安全应用程序开发阶段的器件生命周期状态转换

在以下分析中，以分离式项目开发模型为例。每当分离式项目开发模型和联合式项目开发模型之间存在差异时，均会明确说明。

对于分离式项目开发，在非安全应用程序开发阶段，非安全开发人员会收到处于 NSECSD 状态的 MCU，并进行非安全应用程序开发。在 NSECSD 状态下，只能调试非安全应用程序，安全系统会受到保护以防止非安全开发人员访问。

器件生命周期状态从 NSECSD 更改为 SSD（无需身份验证）

- 如果需要擦除整个闪存以恢复 MCU，非安全开发人员可以擦除整个闪存，以使 MCU 的生命周期状态从 NSECSD 退回到 SSD。但是非安全开发人员需要将 MCU 返回给安全开发人员以重新烧录安全应用程序。

  NSECSD -> SSD：擦除整个闪存

  在 SSD 状态下，非安全开发人员可以使用 RFP 来禁用“**`All erase`**”（全部擦除）命令。
- 联合式开发模型通常不需要这种转换，因为安全和非安全项目的开发通常均在 SSD 状态下进行。

器件生命周期状态从 DPL 退回到 NSECSD（需要身份验证）

- 如表 1 中所述，如果将最终产品的器件生命周期状态设置为 DPL，并且在 NSECSD 状态下已经将 NONSECDBG_KEY 注入 MCU，则最终产品用户可以选择将最终产品返回给非安全开发团队进行故障分析。非安全开发团队可以使用 NONSECDBG_KEY 使器件生命周期状态退回到 NSECSD 状态以进行非安全应用程序调试。

  DPL -> NSECSD：使用 NONSECDBG_KEY 进行身份验证，无需擦除闪存

器件生命周期状态从 NSECSD 更改为 DPL（无需身份验证）

- 在原型设计阶段，非安全开发人员可以将器件生命周期状态从 NSECSD 更改为 DPL，以禁止调试器访问数据和代码。

  NSECSD -> DPL：无需身份验证，无需擦除闪存

2.4 生产流程中的器件生命周期状态转换

分离式项目开发模型和联合式项目开发模型的器件生命周期状态转换如下所述：

分离式项目开发模型

有些大规模生产中器件完成部署后就无法进行调试或更新，在这种情况下，产品制造商通常会执行以下生命周期状态转换：

- 在安全产品制造商处：
  - **CM -> SSD**
  - 烧录安全应用程序
  - **SSD->NSECSD**
- 在非安全产品制造商处：
  - 烧录非安全应用程序
  - **NSECSD -> LCK_DBG**
  - **LCK_DBG -> LCK_BOOT**

联合式项目开发模型

有些大规模生产中器件完成部署后就无法进行调试或更新，在这种情况下，产品制造商通常会执行以下生命周期状态转换：

- 在产品制造商处：
  - **CM -> SSD**
  - 烧录安全和非安全应用程序
  - **SSD -> LCK_DBG**
  - **LCK_DBG -> LCK_BOOT**

扁平化项目开发模型

有些大规模生产中器件完成部署后就无法进行调试或更新，在这种情况下，应用程序团队可以决定保护安全区域中的部分代码。然后，产品制造商采用与联合式项目开发模型相同的生命周期状态转换。

2.5 最终用户可发起的器件生命周期状态转换

最终产品用户可能会收到处于以下状态之一的 MCU：DPL、LCK_DBG 或 LCK_BOOT。

情况 1：最终产品处于 DPL 状态

- 如果需要擦除整个闪存以恢复 MCU，最终用户可以擦除整个闪存以使 MCU 的生命周期状态从 DPL 退回到 SSD。开发可以退回到安全应用程序开发阶段。

  DPL -> SSD：擦除整个闪存，无需身份验证

从 DPL 转换到瑞萨退货申请 (RMA) 状态

- 如表 1 中所述，如果最终产品的器件生命周期状态设置为 DPL，并且需要将最终产品返回给瑞萨进行故障分析，最终用户可以使用 MCU 唯一 ID 将 MCU 的状态从 DPL 更改为 RMA_REQ。最终用户还可以将产品返回给安全开发人员，安全开发人员可以使用 MCU 唯一 ID 或 RAM_KEY（如果之前已安装）将器件状态从 DPL 转换到 RAM_REQ。

  DPL -> RMA_REQ：使用 MCU 唯一 ID 或 RMA_KEY

转换到 RMA_REQ 的注意事项

- 在将生命周期转换到 RMA_REQ 时，将擦除闪存上的全部内容，永久锁定的块或 BPS_SEL 寄存器设置除外。
- 瑞萨在进行故障分析时可以读取永久锁定的块或寄存器中的内容。
- **重要：除非最终用户希望将 MCU 返回给瑞萨进行故障分析，否则请不要执行此转换。**

案例 2：最终产品处于 LCK_BDG 状态

如果最终产品的器件生命周期状态设置为 LCK_DBG，将**永久禁用**器件的调试接口，但串行编程接口仍然可用。但是，串行编程接口无法访问 MCU 代码/数据闪存区域，从而保护最终用户的应用程序。串行编程接口仍然可以提供当前的 MCU 状态，例如引导加载程序版本、器件生命周期状态和 IDAU 区域设置。

案例 3：最终产品处于 LCK_BOOT 状态

如果最终产品的器件生命周期状态设置为 LCK_BOOT，则器件无法退回到任何其他状态。在 LCK_BOOT 状态下，将**永久禁用**调试和串行编程接口，并且器件无法退回到任何其他状态。在决定转换到 LCK_BOOT 状态之前请进行谨慎的评估。

2.6 扁平化项目开发模型的注意事项

使用扁平化项目开发模型时，在开发阶段，用户无需开发具有 Arm® TrustZone® 感知功能的应用程序。在生产过程中，用户可以根据自己的应用选择一个要保护的区域作为安全区域，并相应地设置 IDAU 区域。设置 IDAU 区域后，用户就可以采用与联合式项目开发模型相同的生产流程。有关生产流程的详细信息，请参见第 2.4 节。

3. DLM 密钥创建和安装步骤

3.1 封装密钥安装概述

本节中所提供的信息适用于器件生命周期管理密钥和加密用户密钥。请注意，在*《瑞萨 RA6M4 系列用户手册：硬件》*中，器件生命周期管理密钥和加密用户密钥均称为“用户密钥”。在本应用笔记中，我们以 DLM 密钥为例来讲解安装过程。

要将 DLM 密钥安装到 MCU，需要三个步骤。

1. 客户创建 128 位安装密钥。该密钥称为用户工厂编程密钥 (UFPK)，用于加密用户密钥。客户通过瑞萨密钥封装服务获取封装版本 (W-UFPK) 的密钥。第 3.2 节将引导用户完成封装过程。

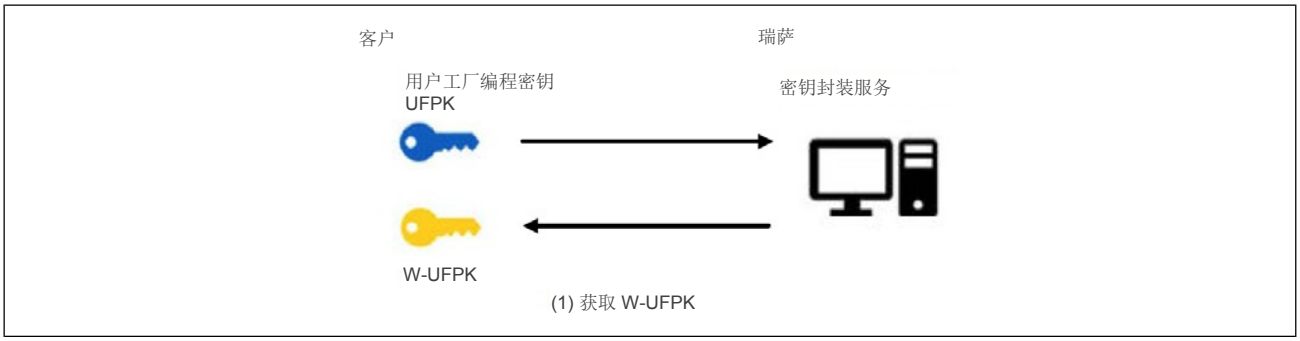

图 14. 封装用户工厂编程密钥 (UFPK)

2. 客户使用 UFPK 作为 AES 密钥来加密用户密钥。第 3.4 节提供的信息可供用户参考和生成加密的 UFPK 密钥。

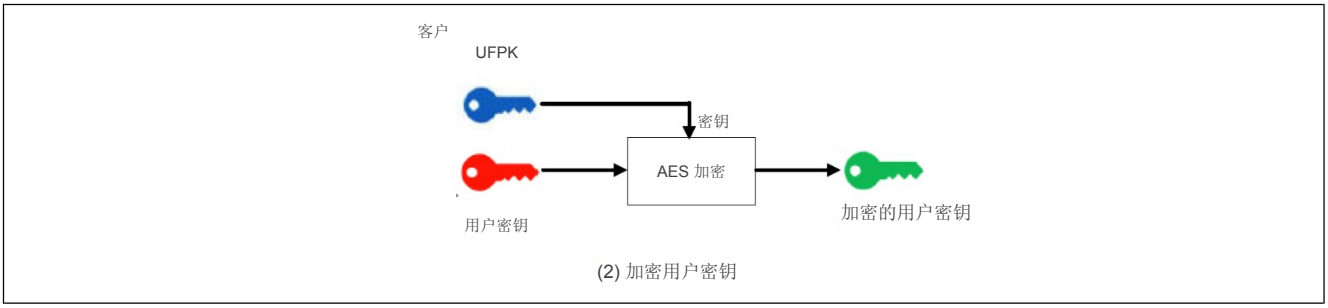

图 15. 使用 UFPK 加密 DLM 密钥

3. 客户通过使用串行编程接口将 W-UFPK（在步骤 1 中生成）和加密的用户密钥（在步骤 2 中生成）发送到 MCU。发送的用户密钥被解密，并使用硬件唯一密钥 (HUK) 封装，然后存储在非易失性存储器中。第 3.5 节将引导用户使用 RFP 完成密钥安装过程。

第 3.6 节将提供有关如何使用 DLM 密钥执行需要经过身份验证的 MCU 器件生命周期转换的信息。

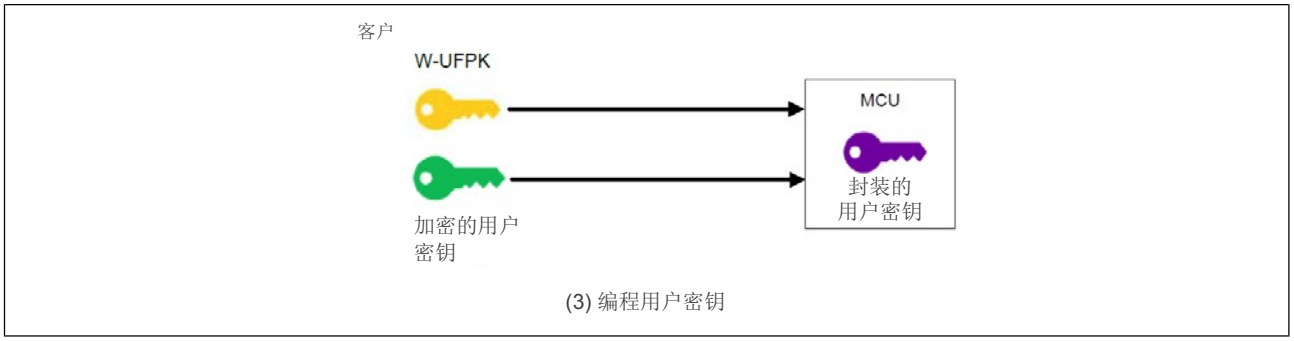

图 16. 创建的封装 DLM 密钥

3.2 创建客户 PGP 密钥对并与瑞萨交换公钥

所有传入和传出 DLM 服务器的信息均使用 PGP 加密，因此客户和瑞萨之间需要交换 PGP 公钥。这是一个一次性过程，需要在与 DLM 服务器建立通信之前完成。

3.2.1 器件生命周期管理 (DLM) 服务器概述

以下是使用 DLM 服务器进行 DLM 密钥封装服务的一般操作流程。用户需要在网络浏览器中登录 https://dlm.renesas.com/ 以访问瑞萨 DLM 服务器。

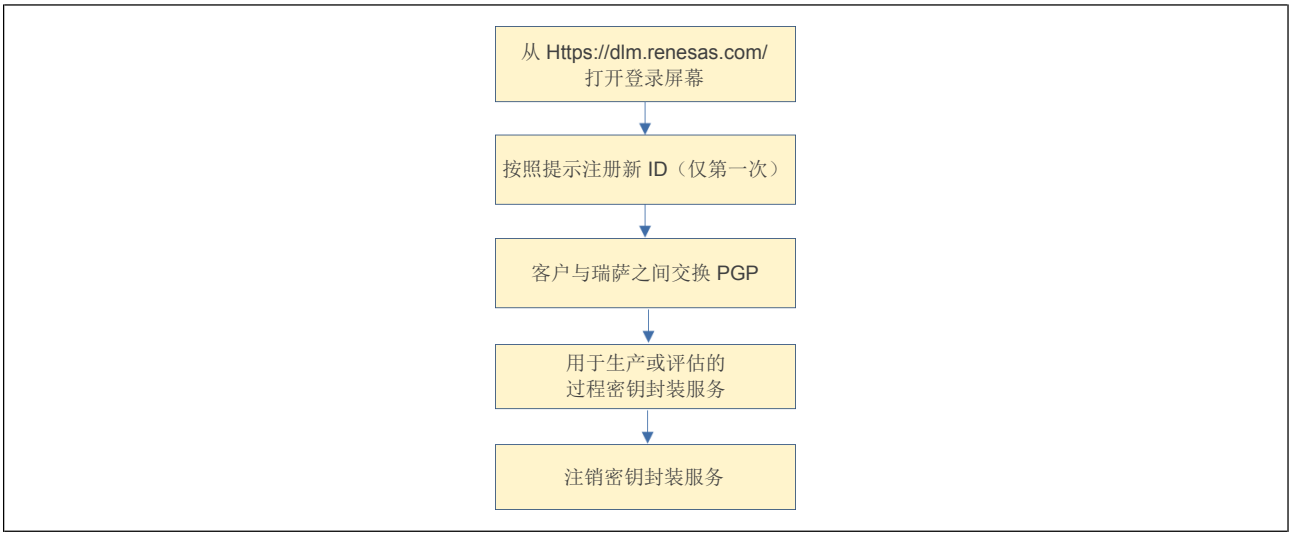

图 17. 使用 DLM 服务器进行密钥封装的操作流程

DLM 服务器常见问题解答 (FAQ) 和用户手册

请注意，打开 https://dlm.renesas.com/ 网页后，用户可以单击右侧的 FAQ链接。用户可以在第一个 FAQ 问题“Is there a manual of this system？”的答案中找到 DLM 服务器用户手册的链接，如图 11 所示。

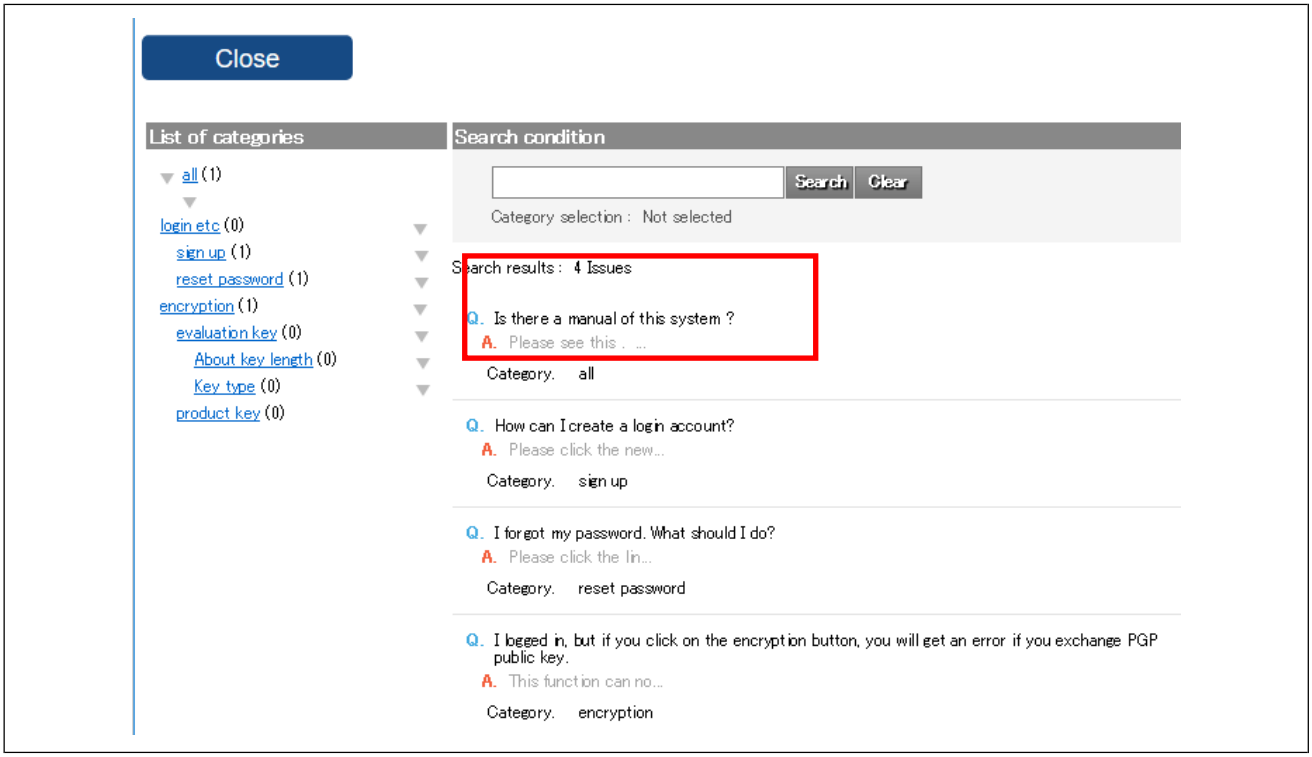

图 18. DLM 服务器常见问题解答 (FAQ) 和用户手册

DLM 服务器 FAQ

如上图所示，FAQ 可以帮助客户解决一些常见问题。

客户和 DLM 服务器之间通信的信息需要经过 OpenPGP 加密。下面会大致介绍密钥封装服务的操作流程，这项服务使用 OpenPGP 加密作为安全措施来保护传输中的 DLM 密钥。

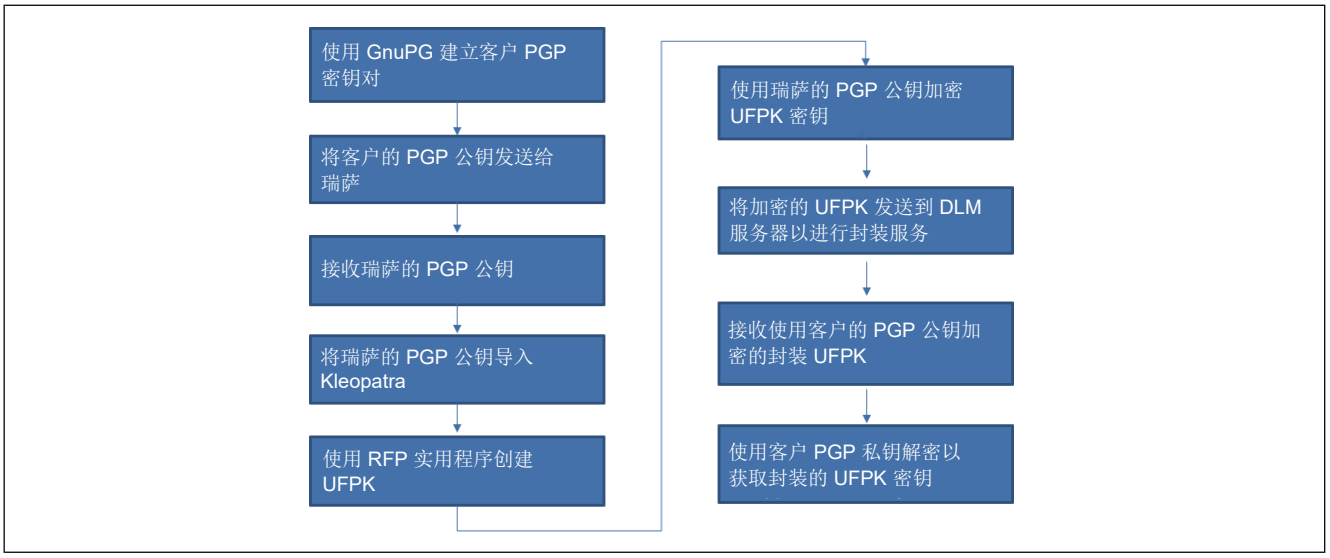

图 19. 使用 PGP 的 DLM 密钥封装服务概述

3.2.2 建立客户 PGP 密钥对

通过以下步骤创建客户 OpenPGP 密钥对。本应用笔记使用官方的 GnuPG Windows® 发行版 Gpg4win 实现 PGP 密钥生成、加密和解密服务。请注意，客户需要采取安全措施来保护 DLM 密钥，以防在使用 KeyWrap 服务时被盗和泄露。

1. 可从以下网站下载 PGP 软件
   http://www.gpg4win.org/

2. 使用 Kleopatra 快捷方式安装和启动该应用程序： 

图 20. 下载 GPG 软件

3. 单击“**File > New Key Pair**”（文件 > 新建密钥对），并选择格式“**Create a personal OpenPGP key pair**”（创建个人 OpenPGP 密钥对）。

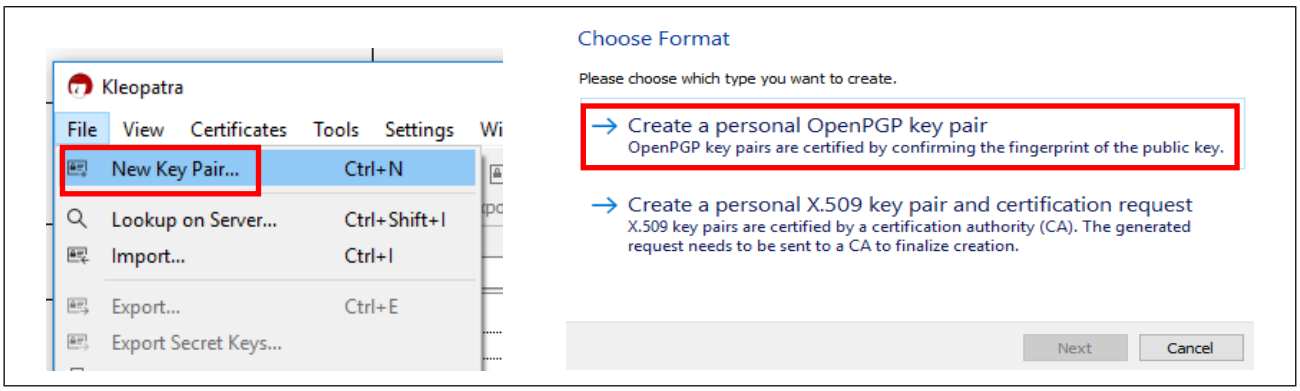

图 21. 生成 PGP 密钥对

4. 单击“**Advanced Settings**”（高级设置）以查看“**Technical Details**”（技术细节）。单击“**OK**”（确定），然后单击“**Next**”（下一步）。

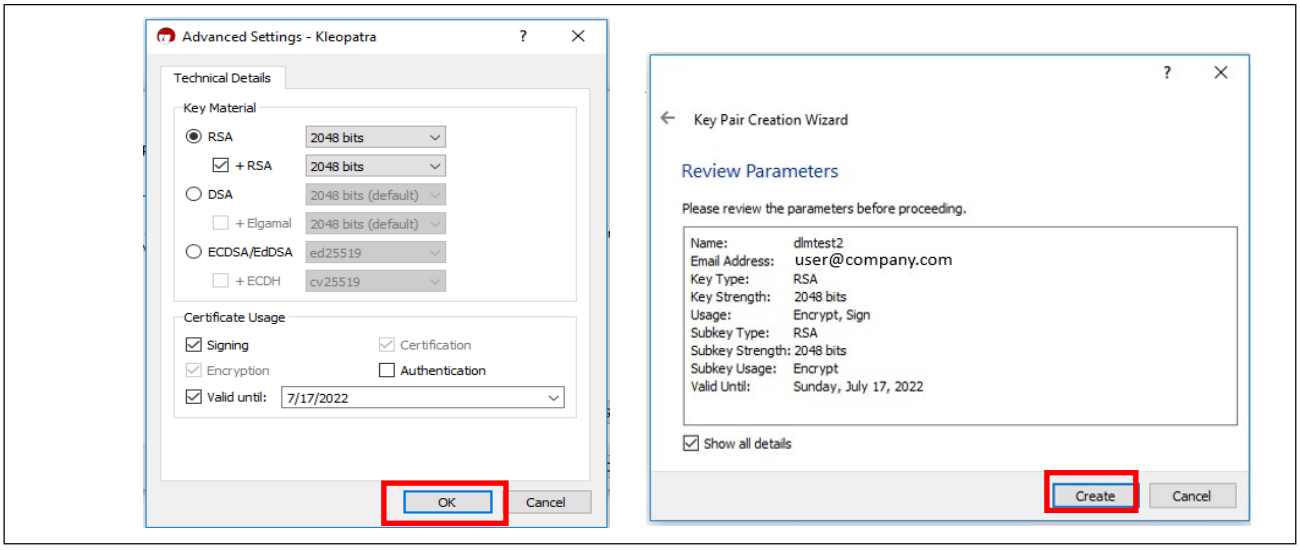

图 22. PGP 密钥对的高级设置

5. 提供密码来保护私钥。确保保存您的密码以备后用。

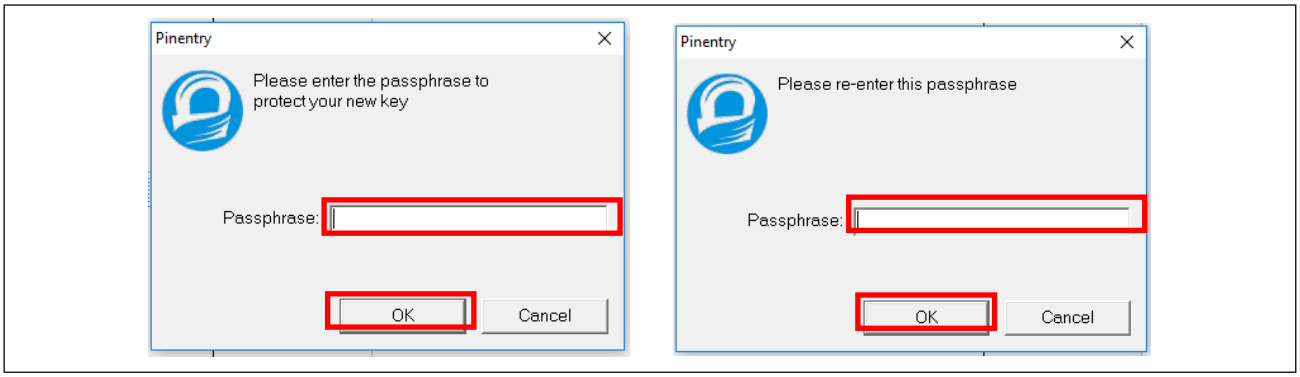

图 23. 提供密码

6. 可以看到 PGP 密钥对创建成功。

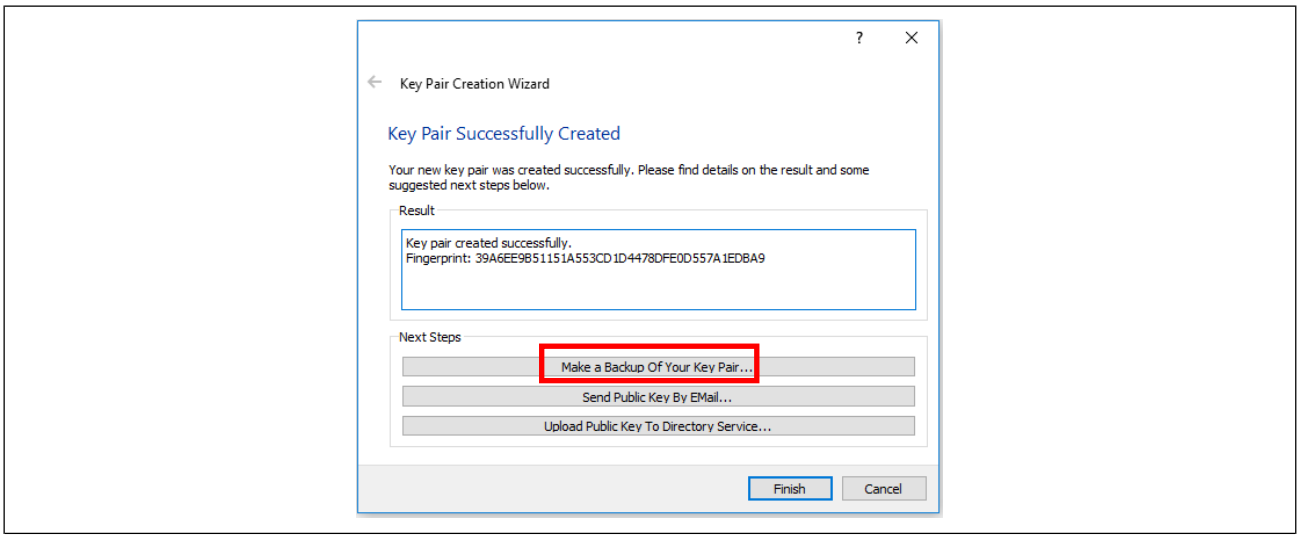

图 24. 密钥对创建成功

7. 使用 Kleopatra 导出客户公钥，如下图所示。

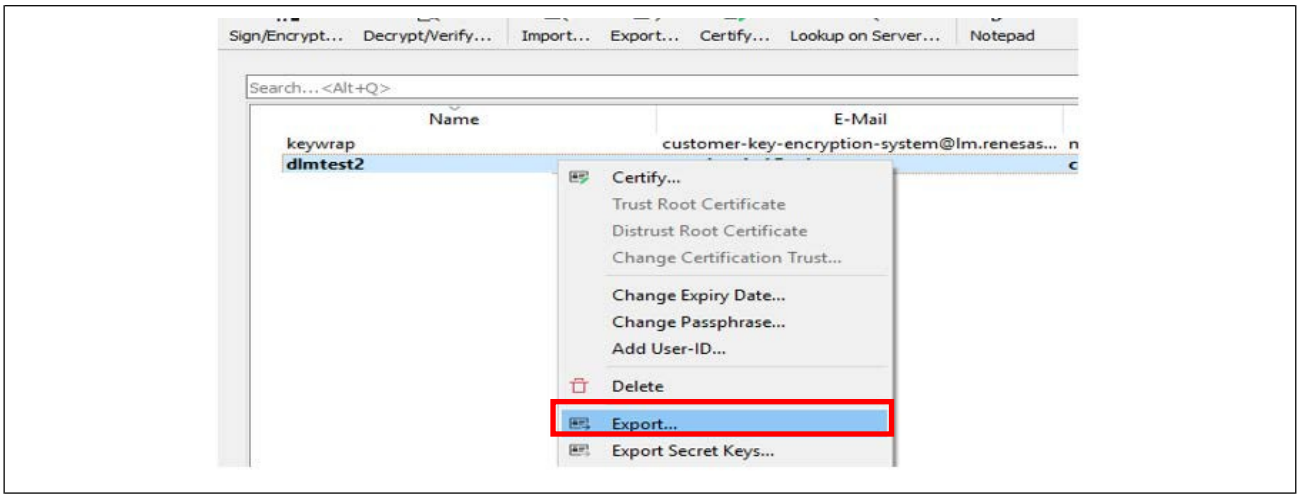

图 25. 导出客户公钥

8. 将客户公钥保存到一个扩展名为 `*.asc` 的文件中，例如“`public_key.asc`”。

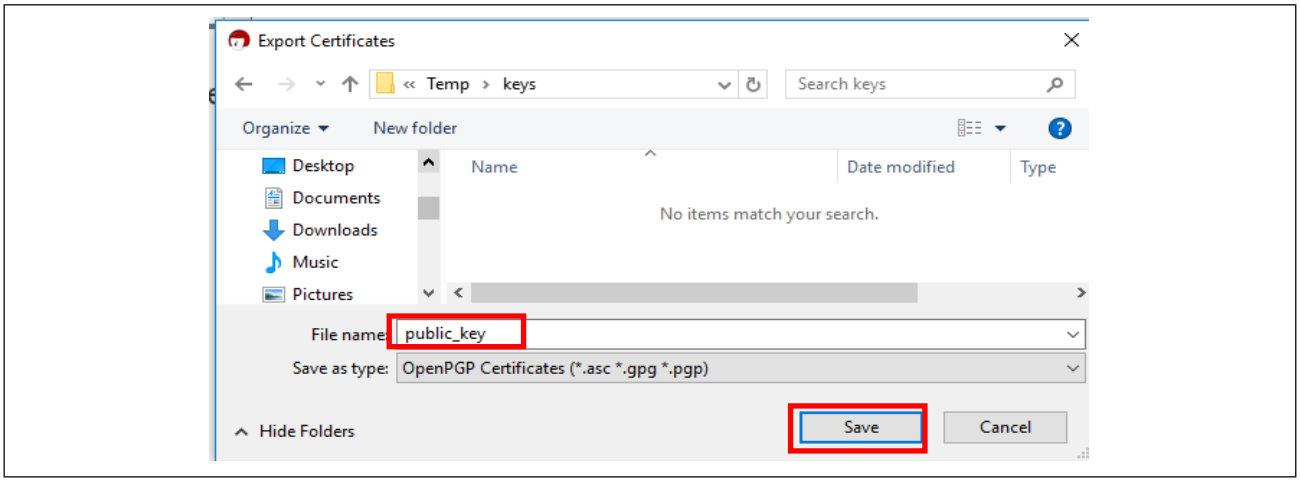

图 26. 保存客户公钥

3.2.3 DLM 服务器注册

本节将简要介绍 DLM 服务器新用户的注册步骤。这是一个面向新客户的一次性过程。有关新用户注册步骤的更多详细信息，建议客户参阅 DLM 用户手册中的*新用户注册*部分。

在浏览器中打开 URL https://dlm.renesas.com/，然后单击“**New registration**”（新用户注册）。按照提示提供电子邮件地址，然后单击“**Send mail**”（发送邮件）。

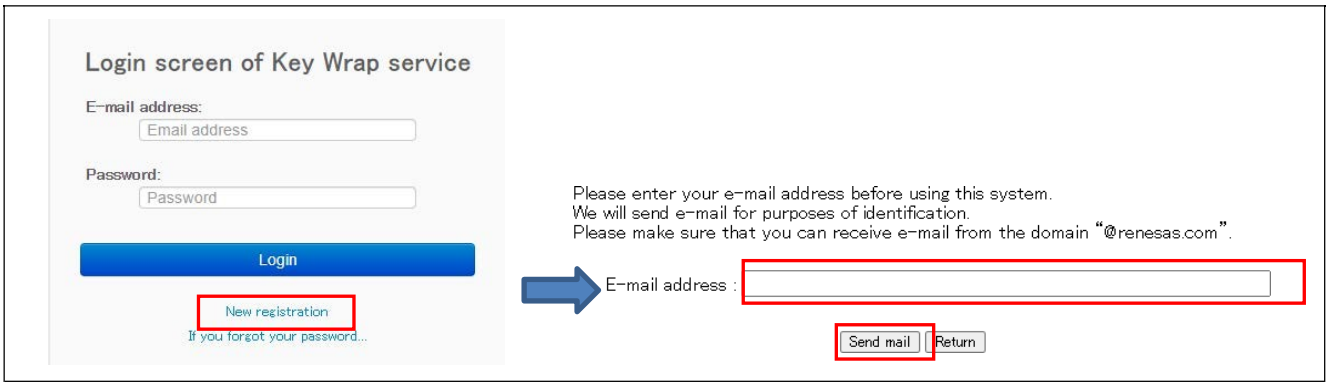

图 27. 提供用于新用户注册的电子邮件地址

您将在电子邮件中收到注册链接，如下图所示。

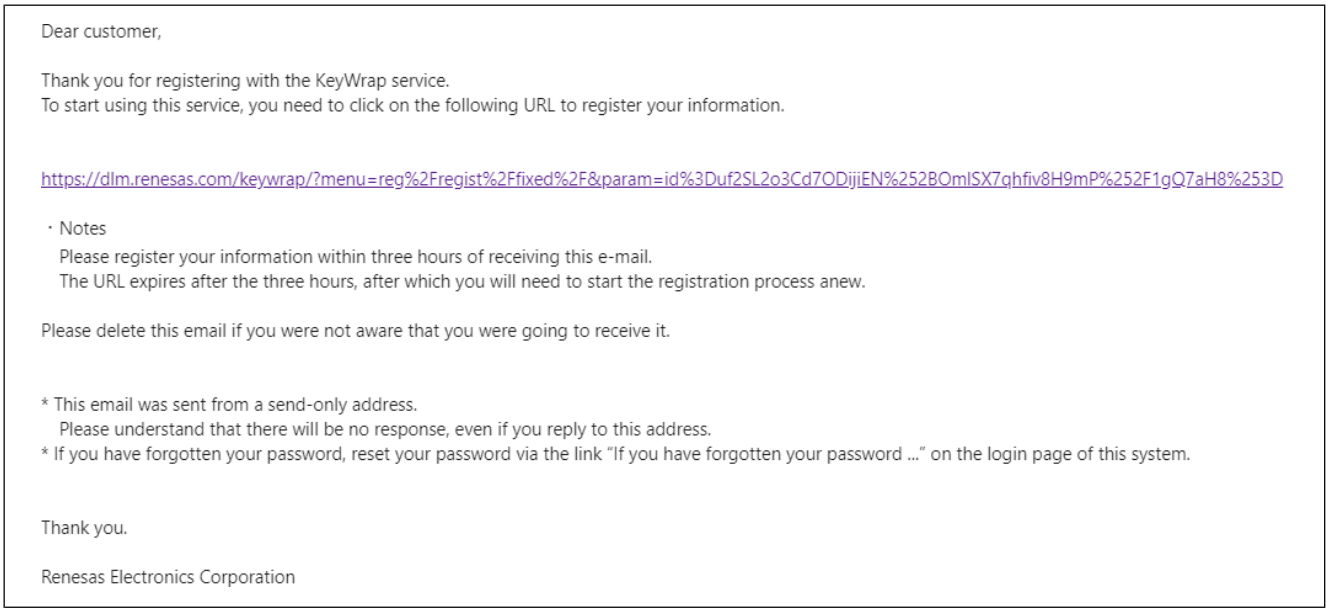

图 28. 内含注册链接的电子邮件

单击确认邮件中的 URL，并提供您的姓名、公司名称和希望设定的登陆密码，然后再次输入登陆密码以供确认。单击“**Next (confirmation)**”（下一步（确认））按钮。显示确认画面后，单击“**Registration**”（注册）按钮完成用户注册。

图 29. 注册客户信息

3.2.4 客户与瑞萨之间交换 PGP 公钥

成功注册客户信息后，以下屏幕随即打开。单击“**Start service**”（开始服务）按钮，开始使用客户的密钥加密系统。

图 30. 开始使用 DLM 服务

单击“**Agree**”（同意）接受“**Trusted Secure IP Key Wrap Agreement**”（受信任安全 IP 密钥封装协议），如下图所示。请注意，每次客户登录 DLM 服务器时都会出现以下协议。

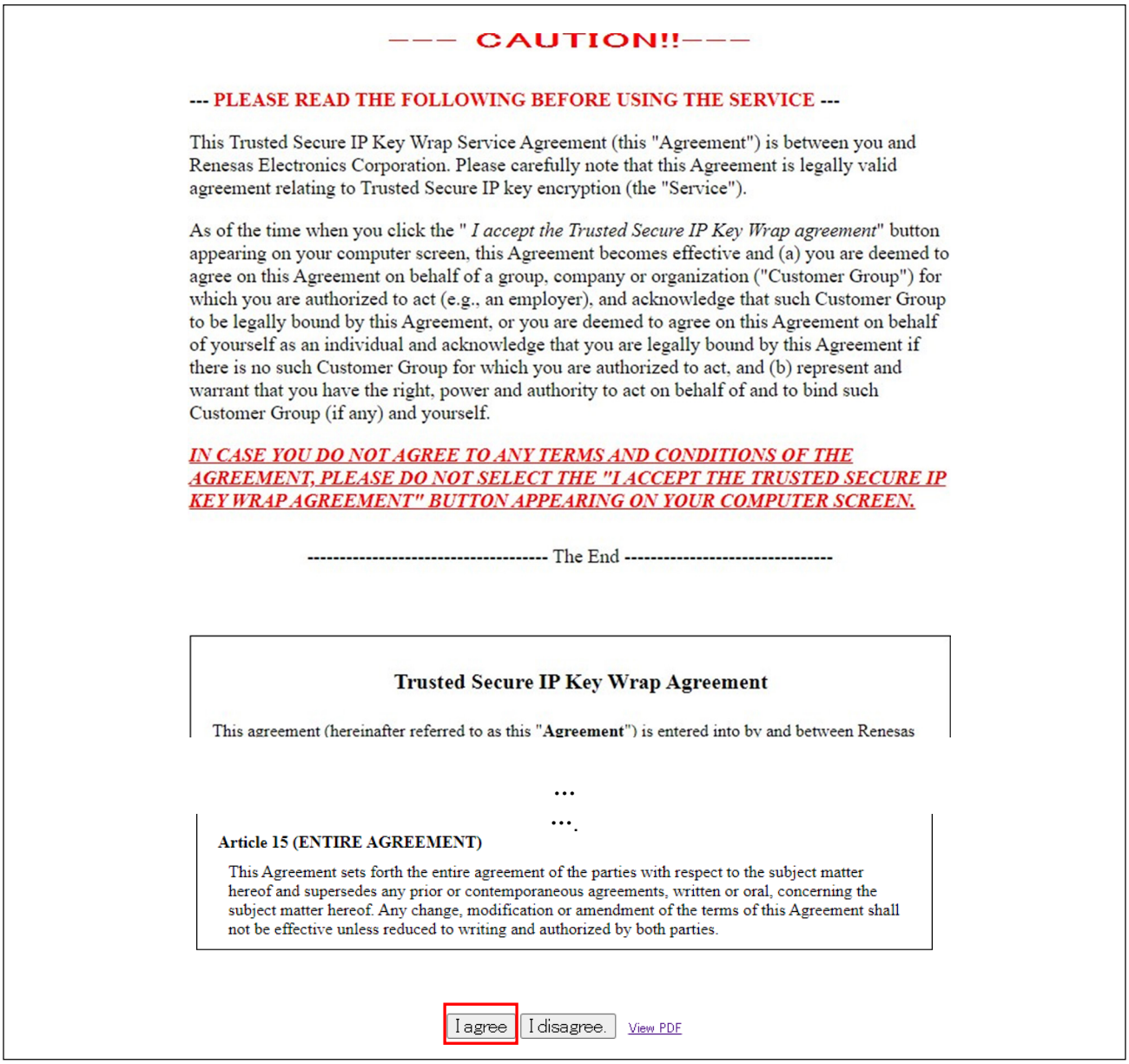

图 31. 同意密钥封装协议

用户和 DLM 服务器之间的通信使用 PGP 来加密所有交换的数据。客户和瑞萨之间需要先进行 PGP 密钥交换，然后才能向 DLM 服务器提供 DLM 密钥。PGP 密钥交换可以通过“**PGP key exchange**”（PGP 密钥交换）按钮实现，如图 32 所示。

在与瑞萨 DLM 服务器交换 PGP 密钥之前，客户登录后将会在红色框中看到如下图所示的消息。PGP 密钥交换成功后，红色文本“**Your PGP key has not been exchanged yet. Start by exchanging your PGP key**”（您的 PGP 密钥尚未交换。请先交换 PGP 密钥）将不会再出现。

图 32. PGP 密钥交换申请

单击“**PGP key exchange**”（PGP 密钥交换）按钮。随即打开如图 33 所示的用户界面。 单击“**Reference**”（引用），如图 33 所示，并选择根据第 3.2.2 节导出的客户公钥 (`public_key.asc`)，如图 26 所示。接下来，在图 33 所示的界面中单击“**PGP key exchange**”（PGP 密钥交换），并等待接收来自新用户注册阶段提供的客户电子邮件地址的瑞萨 PGP 公钥。

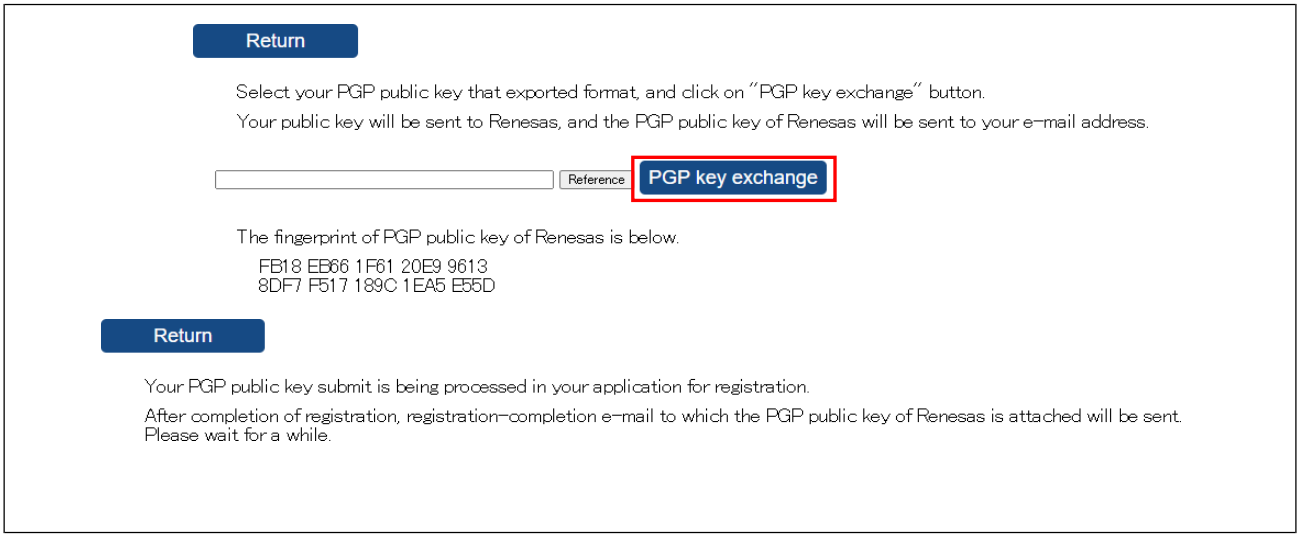

图 33. 向 DLM 服务器提供客户 PGP 公钥

如果注册成功，客户将收到一封电子邮件，内容如图 34 所示。

请注意，PGP 公钥可以注册任意次数。如果多次注册密钥，则会使用最新注册成功的 PGP 公钥进行加密，同时会丢弃所有先前注册的 PGP 公钥。

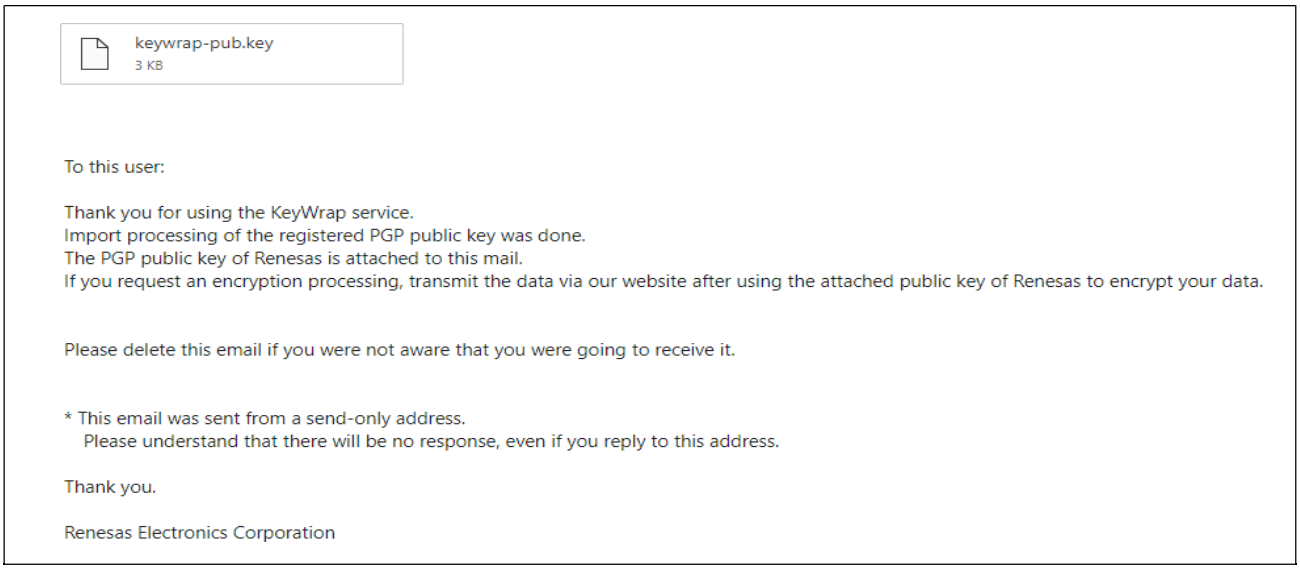

图 34. 接收瑞萨 PGP 公钥

保存收到的瑞萨的 PGP 公钥 (`keywrap-pub.key`)。此密钥将在以下部分中使用。

3.2.5 将瑞萨的 PGP 公钥导入 Kleopatra

返回 Kleopatra 应用程序，并将瑞萨的 PGP 公钥导入 Kleopatra，如下图所示。

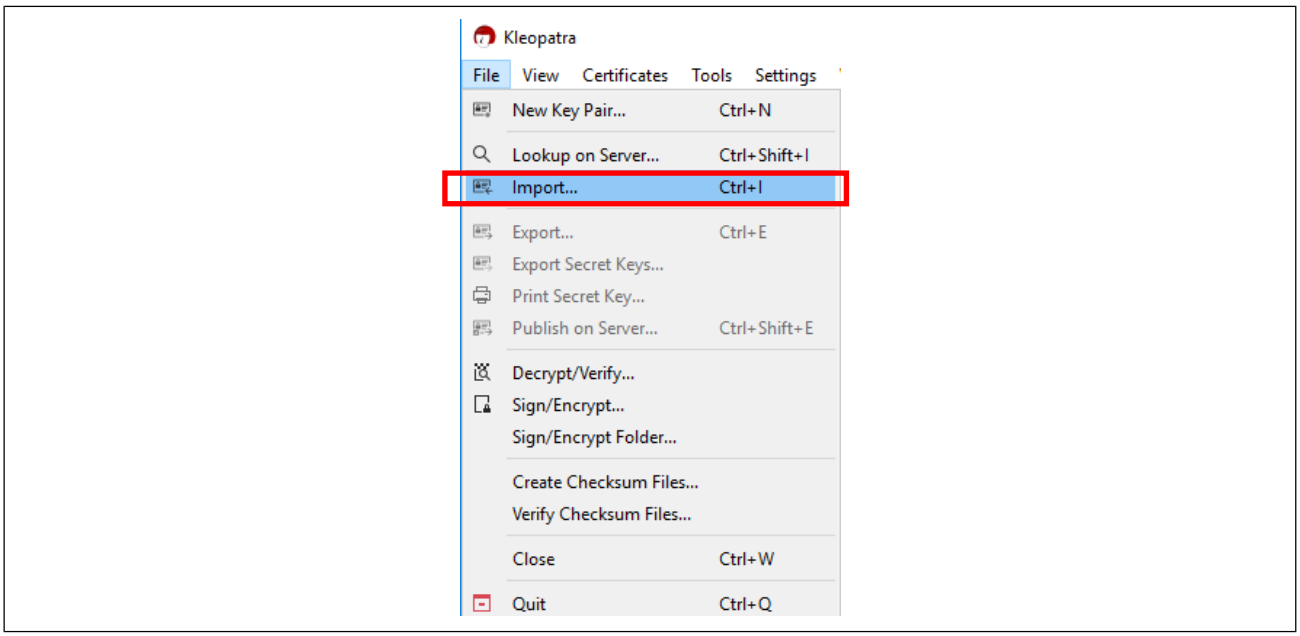

图 35. 将瑞萨公钥导入 Kleopatra

选择上一步保存的 `keywrap-pub.key` 以导入到 Kleopatra。

以下项目将出现在 Kleopatra 中的“Imported Certificates”（已导入证书）中，此时瑞萨公钥即可供使用。

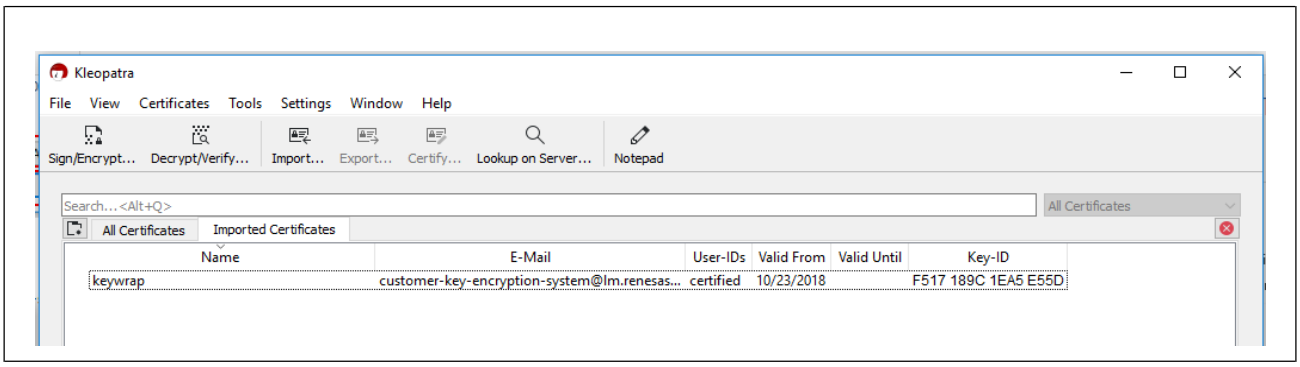

图 36. 瑞萨 PGP 公钥

3.3 使用 RFP 创建 UFPK 和使用 DLM 服务器封装 UFPK

本节将引导用户创建封装的 UFPK，与第 3 章中介绍的 DLM 密钥安装步骤中的前几个步骤相对应。

DLM 密钥只能以封装形式安装到 MCU 上。如第 3.1 节所述，在创建 DLM 密钥之前，需要先创建 UFPK 和封装 UFPK。

- UFPK 由瑞萨 DLM 服务器封装：https://dlm.renesas.com/。该网页通过 HTTPS 和 PGP 与用户通信，从而提供安全措施来保护传输中的用户数据。
- 当 RFP 在 MCU 中安装 DLM 密钥时，需要封装的 DLM 密钥。

3.3.1 创建用户工厂编程密钥 (UFPK)

我们可以使用 RFP 安装文件夹中包含的 `rfp-util.exe` 来生成 UFPK。打开命令行窗口，激活 `rfp-util.exe`，以下图所示内容为例。

命令行输入示例：

```
rfp-util /genufpk /output "C:\DLM_Key_Installation\test\ufpk.key"
```

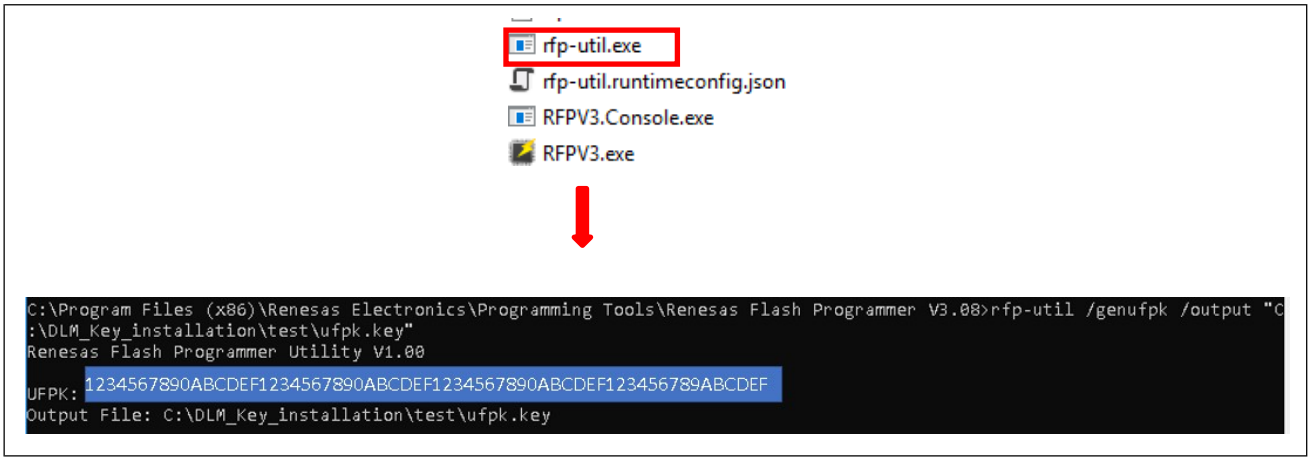

图 37. 生成用户工厂编程密钥

3.3.2 使用瑞萨 PGP 公钥加密 UFPK

从 Kleopatra 中选择“**Sign/Encrypt**”（签名/加密），然后选择使用瑞萨的 PGP 公钥加密此密钥，并使用客户 PGP 密钥签名。

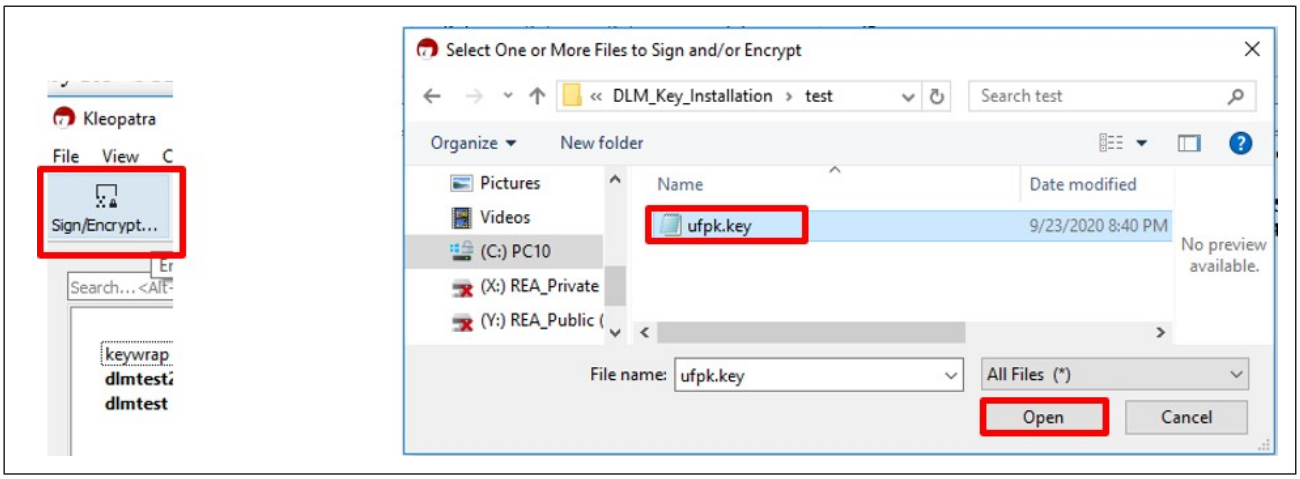

图 38. 选择要加密的 UFPK 密钥

选择“**Encrypt for others**”（为他人加密），并选择瑞萨的 PGP 公钥。单击“**Sign/Encrypt**”（签名/加密）。

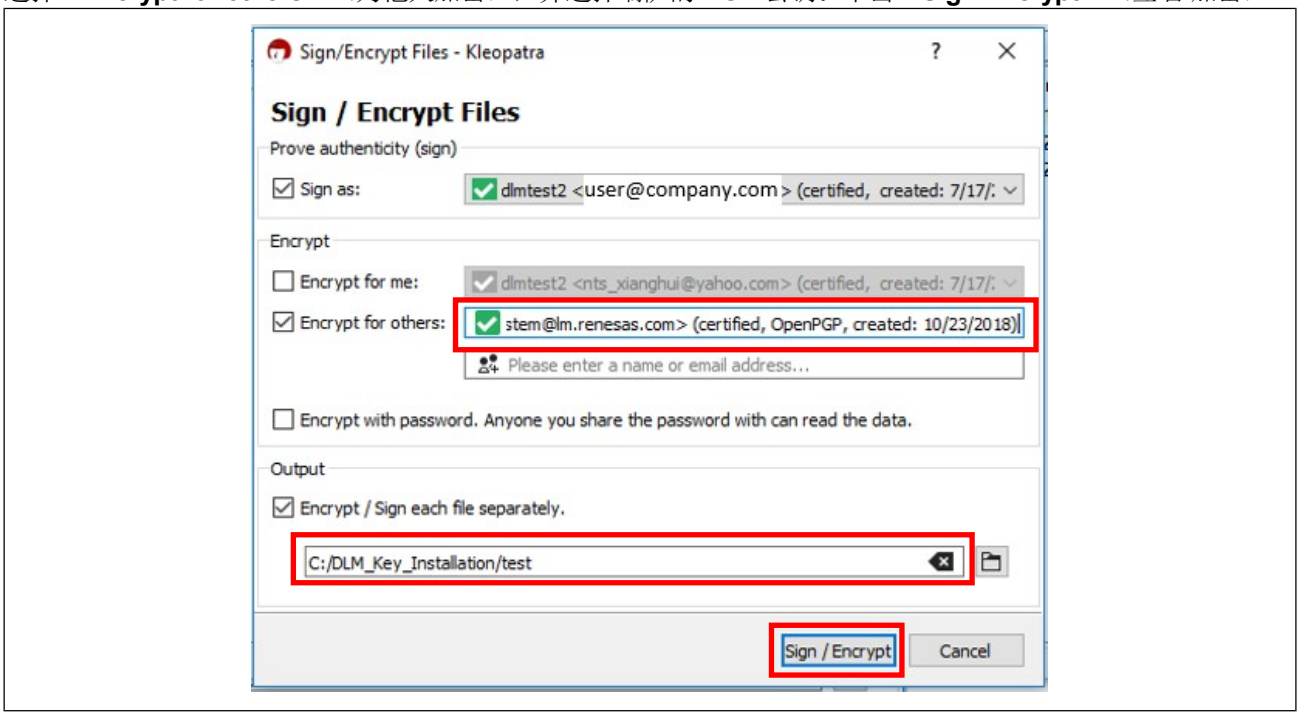

图 39. 使用瑞萨公钥加密 UFPK

您将收到无法解密数据的自我加密警告。按“**Continue**”（继续）。

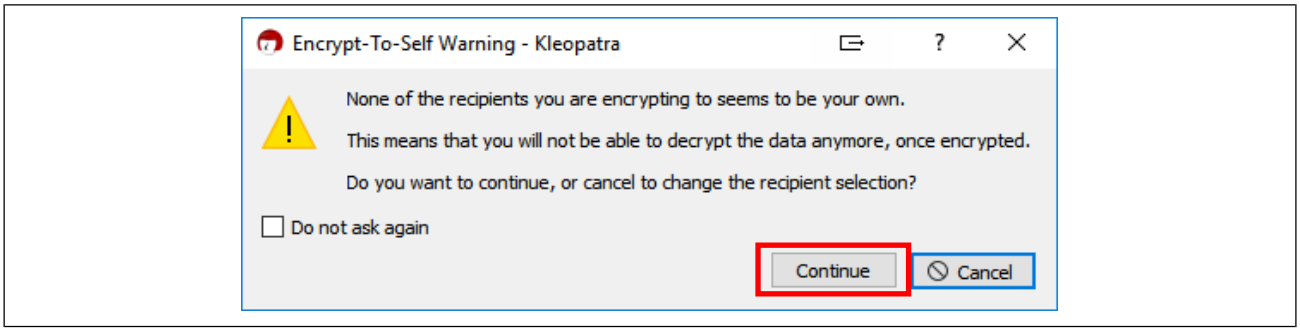

图 40. 确认加密选项

将在所选的文件夹中生成使用瑞萨公钥加密的 UFPK，并添加 .gpg 作为密钥的扩展名。在该用例中，将生成 ufpk.key.pgp。单击“**Finish**”（完成）。

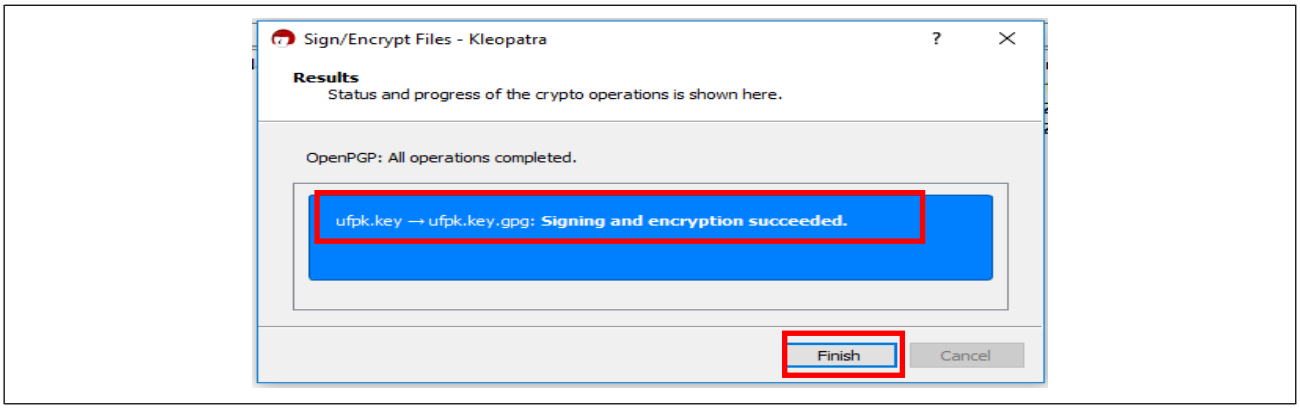

图 41. 使用瑞萨公钥加密 UFPK 密钥

3.3.3 向瑞萨 DLM 服务器发送 UFPK 密钥

现在，我们可以将使用瑞萨公钥加密的 UFPK 发送到瑞萨 DLM 服务器，然后在此由瑞萨私钥解密并由瑞萨 DLM 服务器生成封装的 UFPK (W-UFPK)。

从 DLM 服务器用户界面中，选择 RA 产品家族系列，然后选择“**RA6M4 Encryption of customer's data > Encryption service for products**”（RA6M4 客户数据加密 > 产品加密服务），如下图所示。

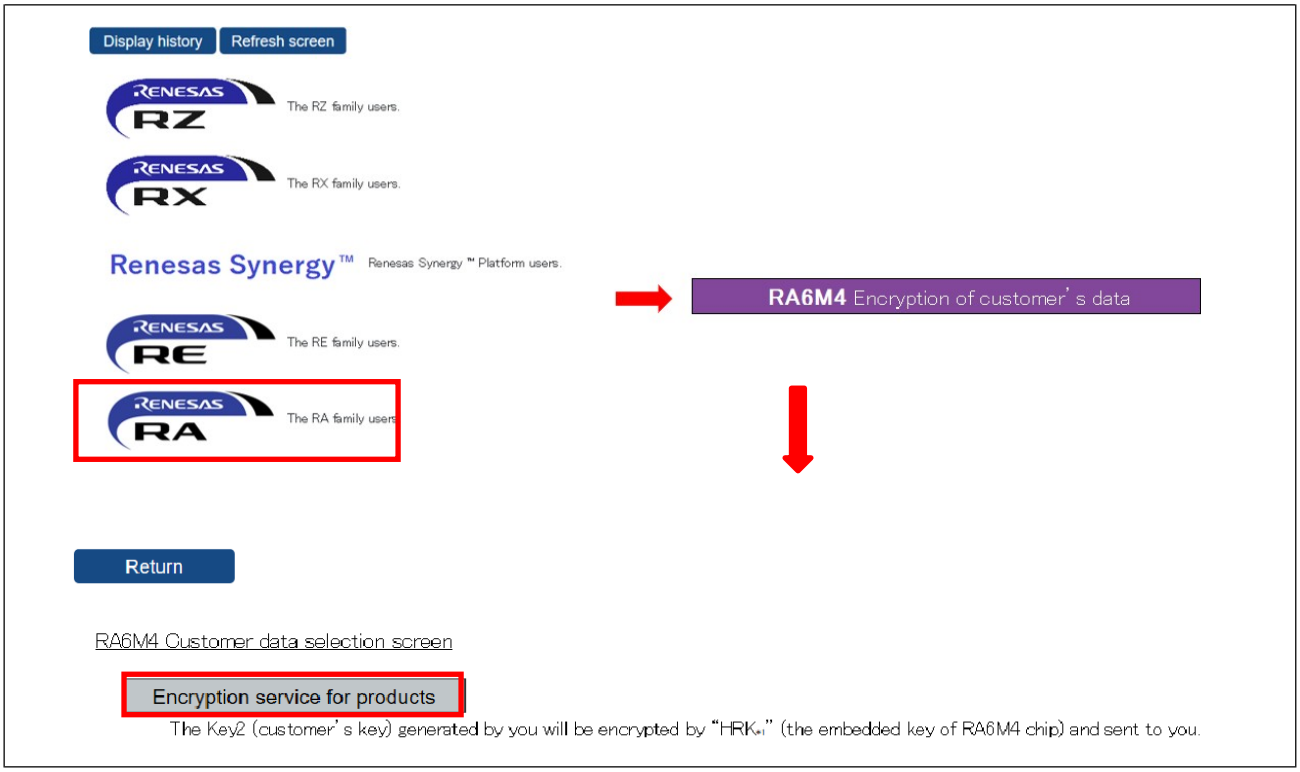

图 42. 选择 RA 器件

接下来，单击“**Reference**”（引用），然后选择根据第 3.3.2 节生成的 .pgp 文件。

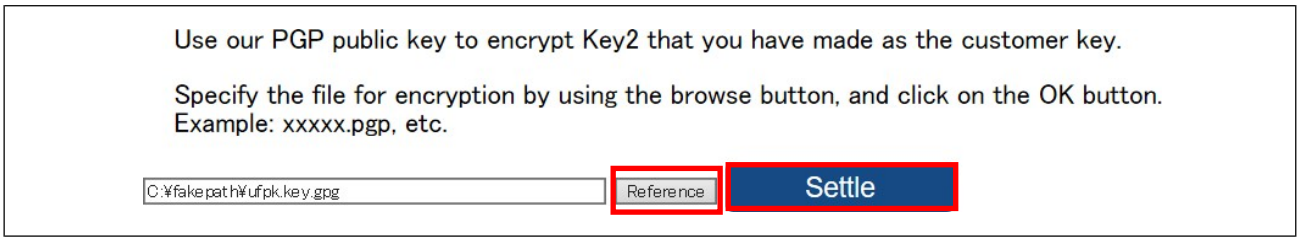

图 43. 将加密的 UFPK 发送到 DLM 服务器

单击“**Settle**”（确定）后会显示以下消息。

图 44. 来自 DLM 服务器的消息

3.3.4 接收使用客户的 PGP 公钥加密的封装 UFPK 密钥

在大多数情况下，使用客户的 PGP 公钥加密的封装 UFPK 密钥应该会在几分钟内发送到您的电子邮箱。

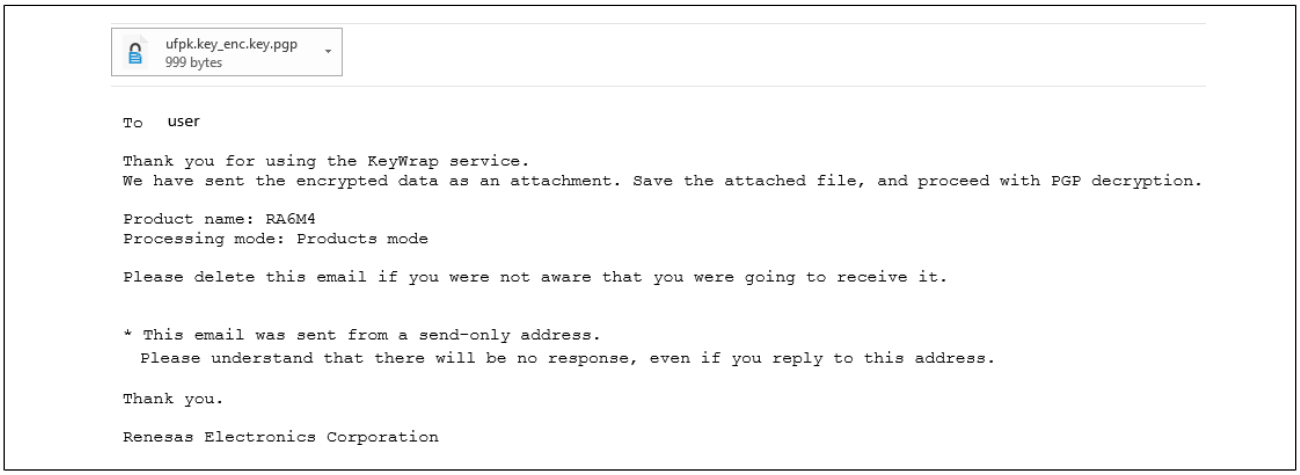

图 45. 接收由瑞萨 PGP 私钥加密的封装 DLM 密钥

下载封装的 UFPK 密钥（由瑞萨私钥加密）以供下一步使用。

3.3.5 使用客户的 PGP 私钥解密加密的封装 UFPK 密钥

通过电子邮件收到的封装 UFPK 已使用客户公钥加密。客户需要使用客户私钥解密此密钥，以获取封装的 UFPK 密钥，然后就可以在 RFP 程序使用它将 DLM Key 安装到 MCU 上。

使用 Kleopatra 程序，单击“**Decrypt/Verify**”（解密/验证），然后选择在第 3.3.4 节收到的封装 UFPK。

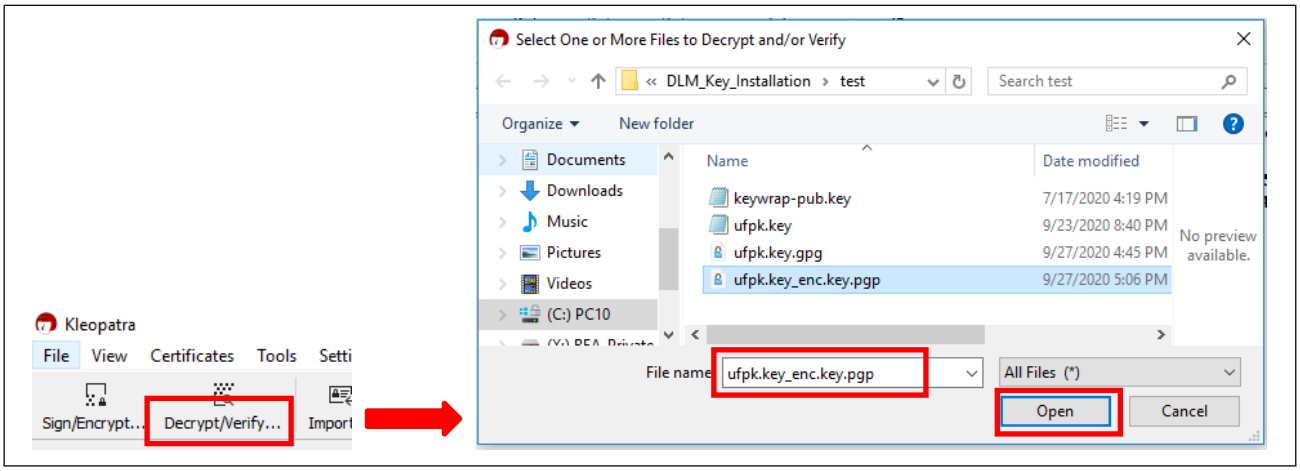

图 46. 使用客户 PGP 私钥解密

接下来，按照提示提供客户 PGP 密钥密码。

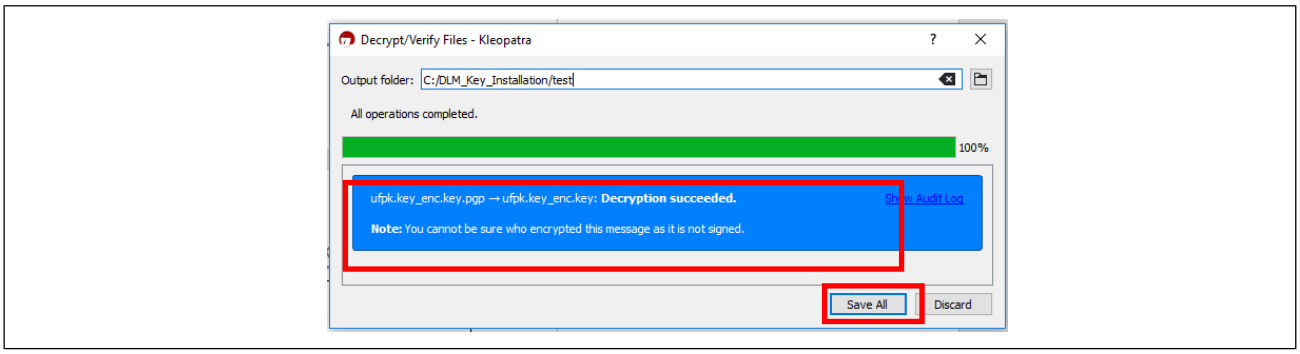

图 47. 使用客户 PGP 私钥解密

3.4 生成使用 UFPK 和 WUFPK 加密的 DLM 密钥

如图 16 所示，为创建封装 DLM 密钥，DLM 密钥需要由 UFPK 加密，并且加密的 DLM 密钥和 W-UFPK 均需要通过串行编程接口发送到 MCU 以创建封装 DLM 密钥。

RFP 命令行工具 rfp-util.exe 可用于生成密钥捆绑包，然后要使用 RFP 将捆绑包安装到 MCU 里。用户应使用 rfp-util.exe 及 UFPK 和 W-UFPK 作为输入以生成供 RFP 使用的密钥捆绑包。下面是用于生成要安装在 RA6M4 MCU 上的 SECDBG_KEY 的命令行输入示例。

命令行输入示例：

```
rfp-util /genkey /ufpk "C:\DLM_Key_Installation\test\ufpk.key" /wufpk
"C:\DLM_Key_Installation\test\ufpk.key_enc.key" /key
"000102030405060708090A0B0C0D0E0F" /output
"C:\DLM_Key_Installation\test\SECDBG.rkey"
```

请注意，如果 SECDBG.rkey 安装在处于 SSD 状态的 MCU 上，则 **000102030405060708090A0B0C0D0E0F** 就是用于使 MCU 器件生命周期状态从 NSECSD 退回到 SSD 的明文 DLM 密钥数据。客户可以身份验证数据（DLM密钥）。必须确保此信息安全，不得泄露。

下面是生成的 SECDBG 密钥文件示例，可以使用 RFP 安装到 MCU。

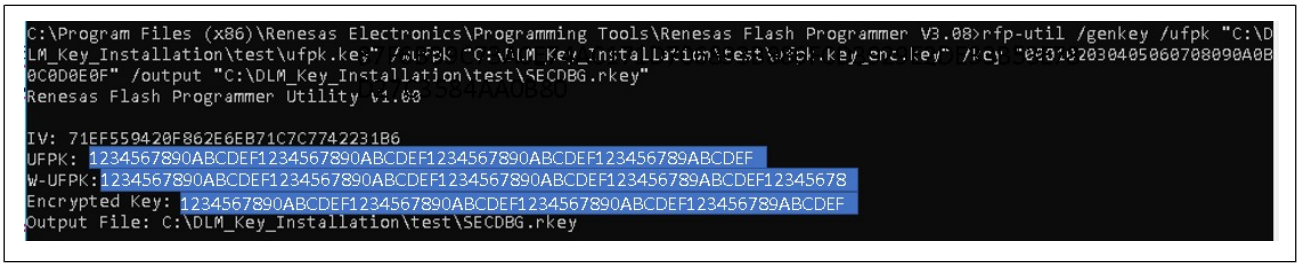

图 48. 使用 rfp-util.exe 生成 DLM 密钥 SECDNG_KEY

同样，用户可以使用以下命令行输入生成 NONSECDBG_KEY：

```
rfp-util /genkey /ufpk "C:\DLM_Key_Installation\test\ufpk.key" /wufpk
"C:\DLM_Key_Installation\test\ufpk.key_enc.key" /key
"010102030405060708090A0B0C0D0E0F" /output "C:\DLM_Key_Installation\test\NON-
SECDBG.rkey"
```

请注意，如果 NONSECDBG.rkey 安装在处于 NSECSD 状态的 MCU 上，则 **010102030405060708090A0B0C0D0E0F** 就是用于使 MCU 器件生命周期状态从 DPL 退回到 NSECSD 的明文 DLM 密钥数据。客户可以自定义用于其应用程序的身份验证数据（DLM密钥）。必须确保此信息安全，不得泄露。

下面是生成的 NONSECDBG 密钥文件示例，可以使用 RFP 安装到 MCU。

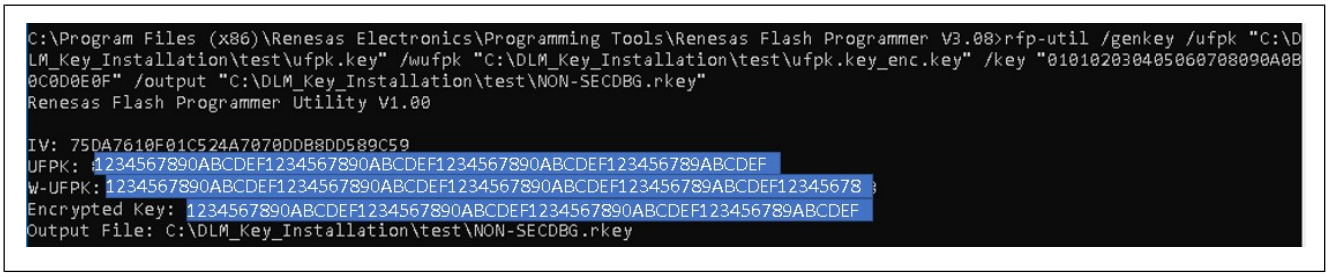

图 49. 使用 rfp-util.exe 生成 DLM 密钥 NONSECDBG_KEY

同样，用户可以使用以下命令行输入生成 RMA_KEY：

```
rfp-util /genkey /ufpk "C:\DLM_Key_Installation\test\ufpk.key" /wufpk
"C:\DLM_Key_Installation\test\ufpk.key_enc.key" /key
"020102030405060708090A0B0C0D0E0F" /output
"C:\DLM_Key_Installation\test\RMA.rkey"
```

请注意，如果 RMA.rkey 安装在处于 SSD 状态的 MCU 上，则 **02010203040506070809 0A0B0C0D0E0F 就**是用于使 MCU 器件生命周期状态从 DPL 前进到 RMA_REQ 的明文 DLM 密钥数据。客户可以自定义用于其应用程序的身份验证数据（DLM 密钥）。必须确保此信息安全，不得泄露。

下面是生成的 RMA 密钥文件示例，可以使用 RFP 安装到 MCU。

```
C:\Program Files (x86)\Renesas Electronics\Programming Tools\Renesas Flash Programmer V3.08>rfp-util /genkey /ufpk "C:\D
LM_Key_Installation\test\ufpk.key" /wufpk "C:\DLM_Key_Installation\test\ufpk.key_enc.key" /key "020102030405060708090A0B
0C0D0E0F" /output "C:\DLM_Key_Installation\test\RMA.rkey"
Renesas Flash Programmer Utility V1.00

IV: 6C329F7584D59BD359B736FC8BA7CB49
UFPK: 1234567890ABCDEF1234567890ABCDEF1234567890ABCDEF123456789ABCDEF
W-UFPK: 1234567890ABCDEF1234567890ABCDEF1234567890ABCDEF123456789ABCDEF12345678
Encrypted Key: 1234567890ABCDEF1234567890ABCDEF1234567890ABCDEF123456789ABCDEF
Output File: C:\DLM_Key_Installation\test\RMA.rkey
```

图 50. 使用 rfp-util.exe 生成 DLM 密钥 RMA_KEY

用户现在可以继续使用 RFP 安装生成的 DLM 密钥（SECDBG.rkey、RMA.rkey 和 NONSECDBG.rkey）。

3.5 DLM 密钥安装

本节将提供执行 DLM 密钥安装所需的配置设置。有关如何创建 RFP 项目和建立与目标板的连接的说明，请参见《RFP 用户手册》。本节将提供每种状态转换的关键配置设置。

对于不进行身份验证的转换和“**All erase**”（全部擦除）操作，请参见第 1.2.5 节。

基于 Arm® TrustZone® 技术的 RA MCU 包括一个 256 位硬件唯一密钥（HUK，请参见第 5.1 节），HUK 是在 MCU 的 CM 状态期间安装的。每个 MCU 均具有唯一的 HUK，并可确保 DLM 密钥的安全存储。DLM 密钥将通过 HUK 封装后安装到 MCU 里。此安装过程可确保 DLM 密钥仅在安装的 MCU 上可用。

3.5.1 安装安全调试密钥

客户可以选择在 SSD 状态下安装安全调试密钥 (SECDBG_KEY) 和退货授权 (RAM_KEY)。在安装密钥之前，确保满足以下两个条件：

- MCU 处于 SSD 状态
- SECDBG_KEY 和 RMA_KEY 是按照第 3.4 节中的步骤生成的

要安装安全调试密钥和退货授权密钥，首先将 DLM 密钥上传到 RFP 程序。本节以上一节生成的安全调试密钥为例。同样的步骤也适用于安装退货授权密钥。

解压缩 ra6m4_dlm_key_install.rfp.zip 以显示 ra6m4_dlm_key_install.rfp.rpj。启动 RFP，然后打开项目 ra6m4_dlm_key_install.rfp.rpj，并查看本节中介绍的设置。

在“**Flash Options**”（闪存选项）中，使用“**Set Option**”（设置选项）来设置“**Set**”（设置）。现在，突出显示“**Encrypted SECDBG Key**”（加密的 SECDBG 密钥），然后单击右侧的“**...**”来选择对应的 SECDBG_KEY。

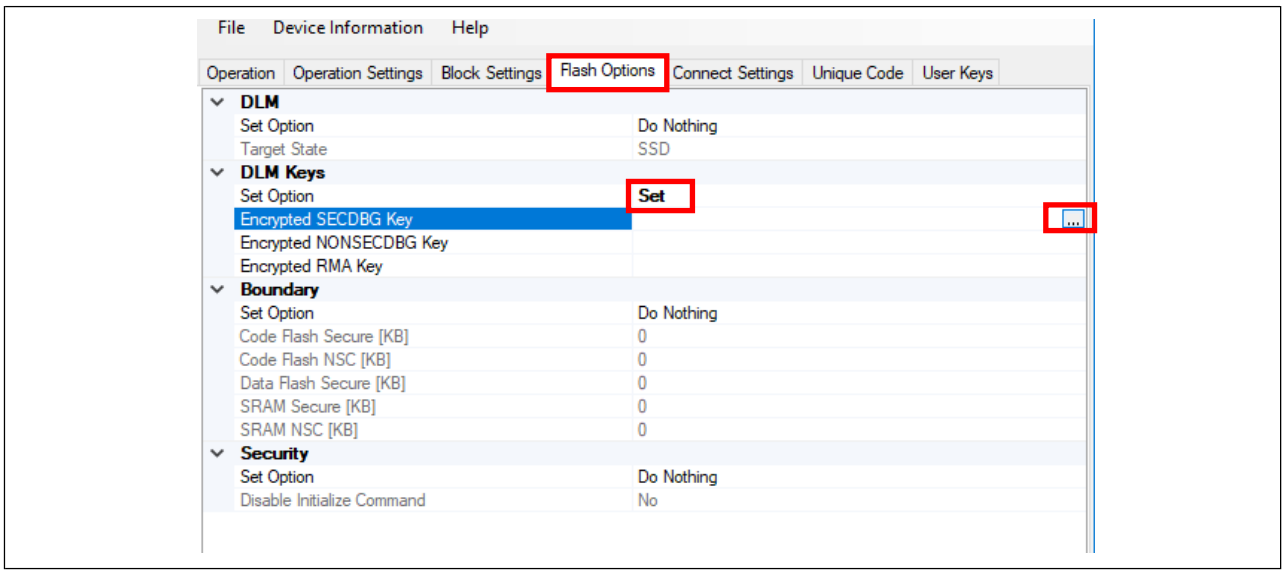

图 51. 选择要安装的 SECDBG_Key

选择第 3.4 节生成的 `SECDBG.rkey`。

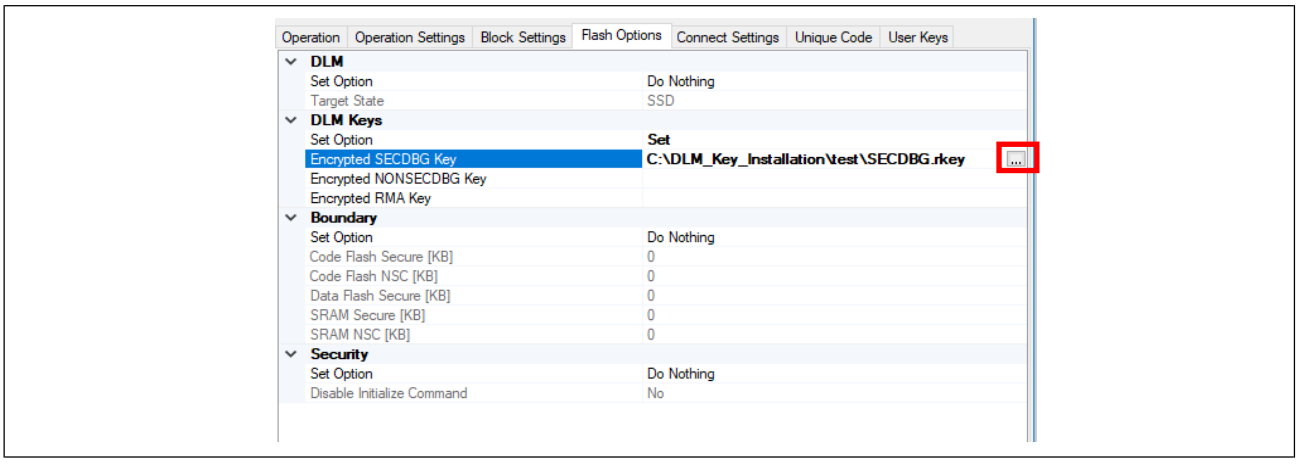

图 52. 选择要安装的 SECDBG_KEY

在“**Operation Settings**”（操作设置）选项卡下，选择“**Program Flash Options**”（烧录闪存选项）和“**Verify Flash Options**”（验证闪存选项）。

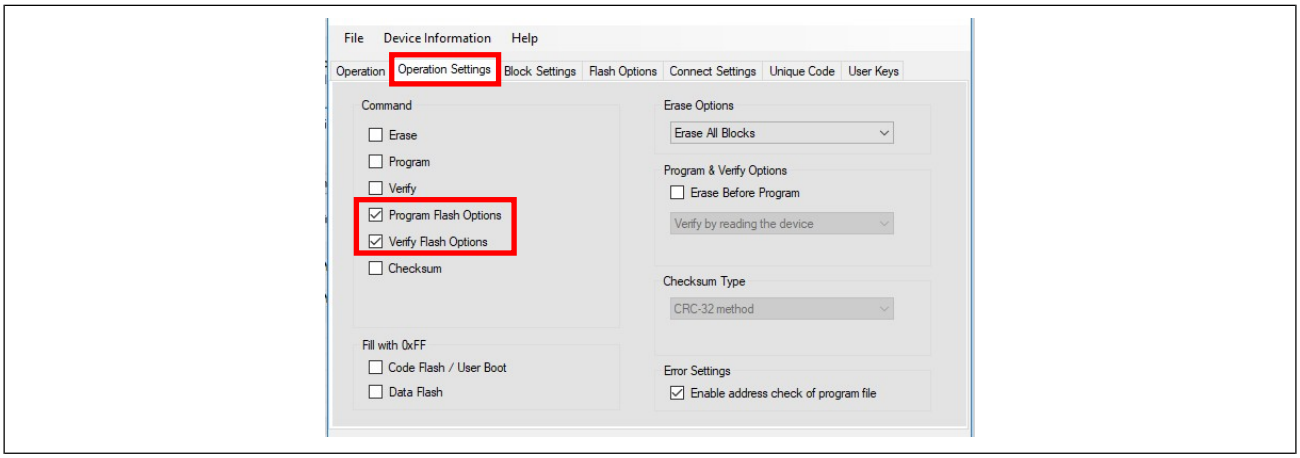

图 53. 选择烧录和验证闪存选项设置用于设置 DLM 密钥的 RFP 操作可以在“Operation”（操作）选项卡下烧录所选的有效二进制文件的同时完成。可以选择在下载文件中包含二进制文件。

导航到“**Operation**”（操作）选项卡，选择要使用的安全应用程序二进制文件，然后单击“**Start**”（开始）烧录设置。

解压缩 `test.zip` 以显示 `bare_metal_minimal_s.srec` 和 `bare_metal_minimal_ns.srec`。出于测试目的，本应用笔记中包含了这些二进制文件，以便于用户参考。

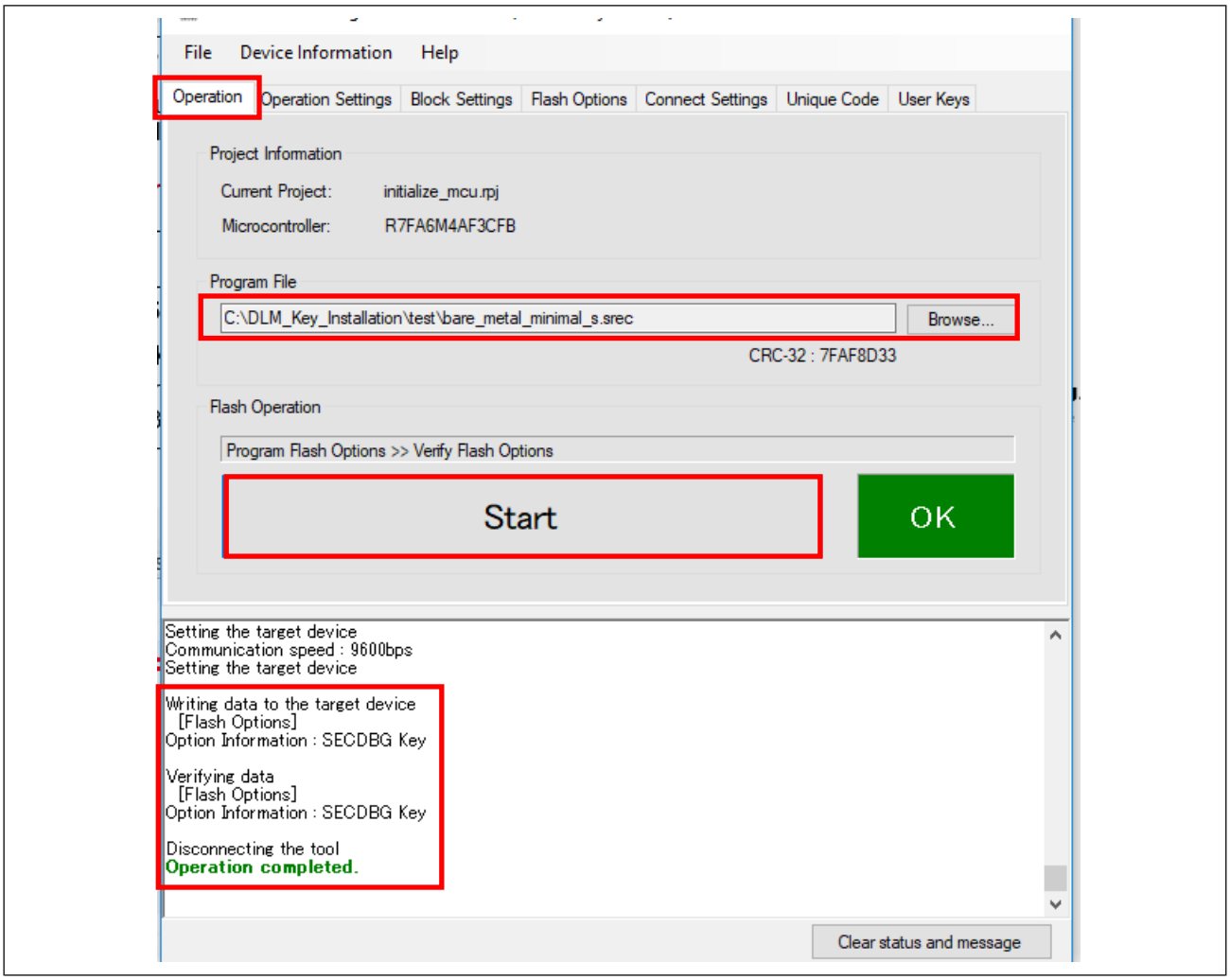

图 54. 安装 SECDBG_KEY

3.5.2 安装非安全调试密钥

在安装非安全调试密钥之前，将 MCU 器件生命周期状态转换到 NSECSD。

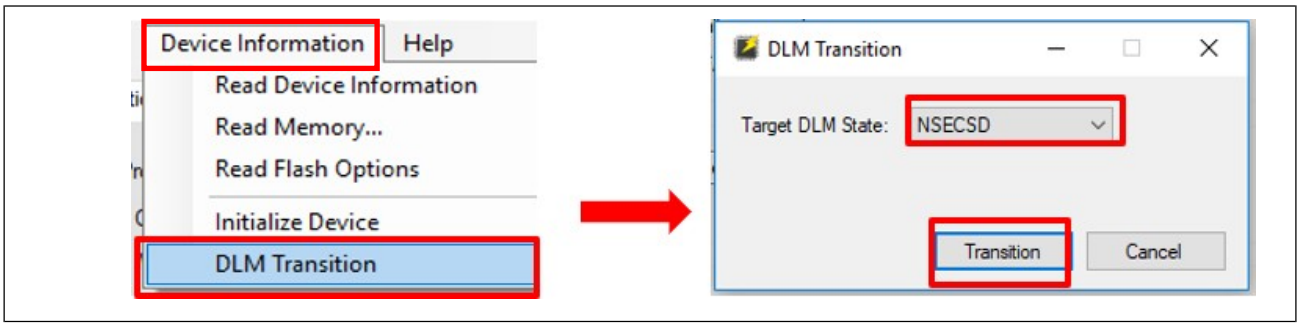

图 55. 将 MCU 器件生命周期状态转换到 NSECSD

接下来，按照类似于第 3.5.1 节的步骤安装非安全调试密钥。在此示例中，我们可以使用图 49 生成的 NONSECDBG_KEY (`NONSECDBG.rkey`) 来说明操作。

请注意，用户需要删除 SECDBG 密钥文件条目，并添加 NONSECDBG 密钥文件条目，如下图所示。

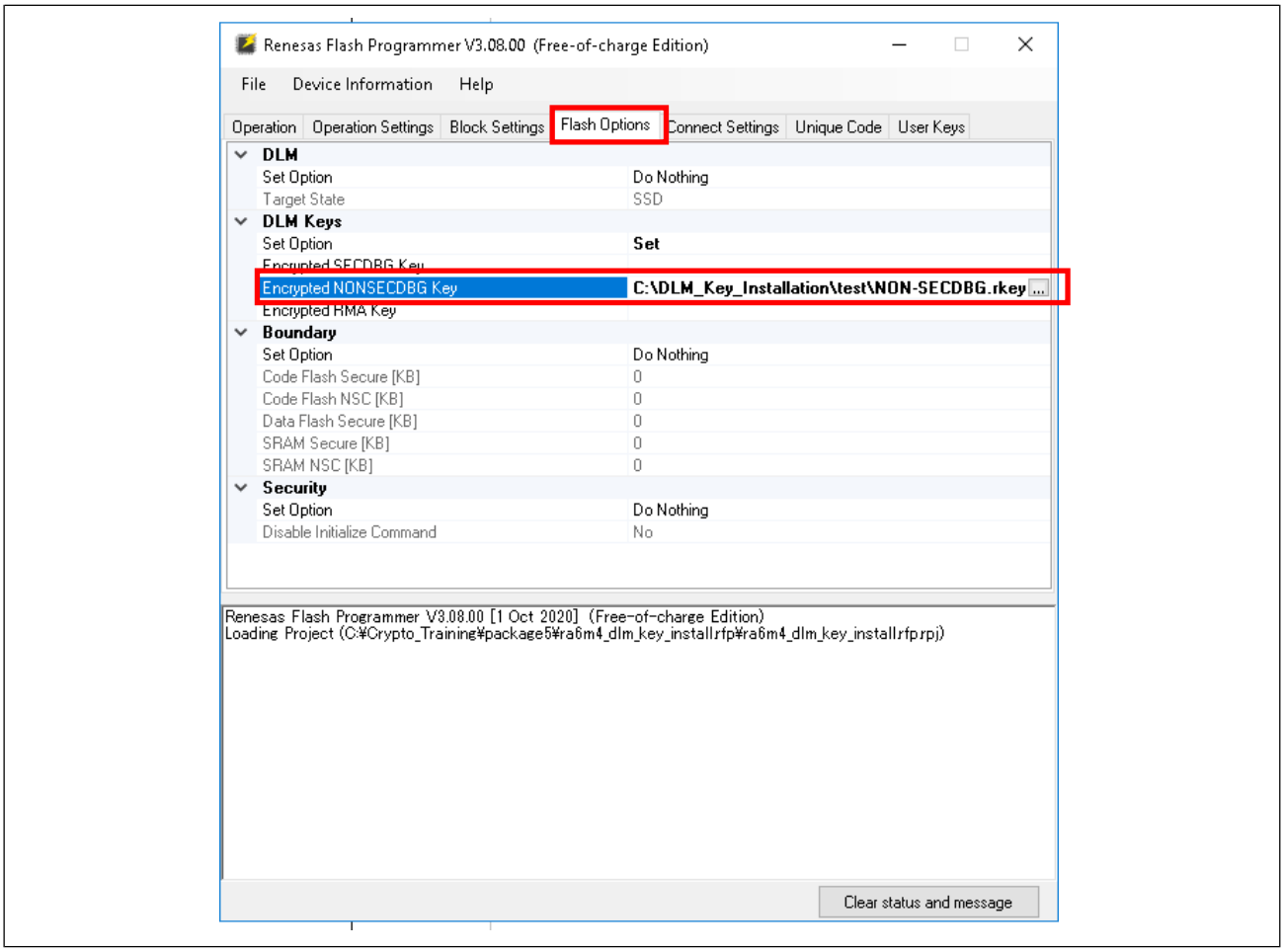

图 56. 选择要安装的 NONSECDBG.rkey

与安装 SECDBG_KEY 类似，设置 DLM 密钥的 RFP 操作可以在“**Operation**”（操作）选项卡下烧录所选的有效非安全二进制文件的同时完成。可以选择安装带有 DLM 密钥安装的二进制文件。

如前文所述，出于测试目的，本应用笔记中包含了 `bare_metal_minimal_ns.srec` 二进制文件，以便于用户参考。

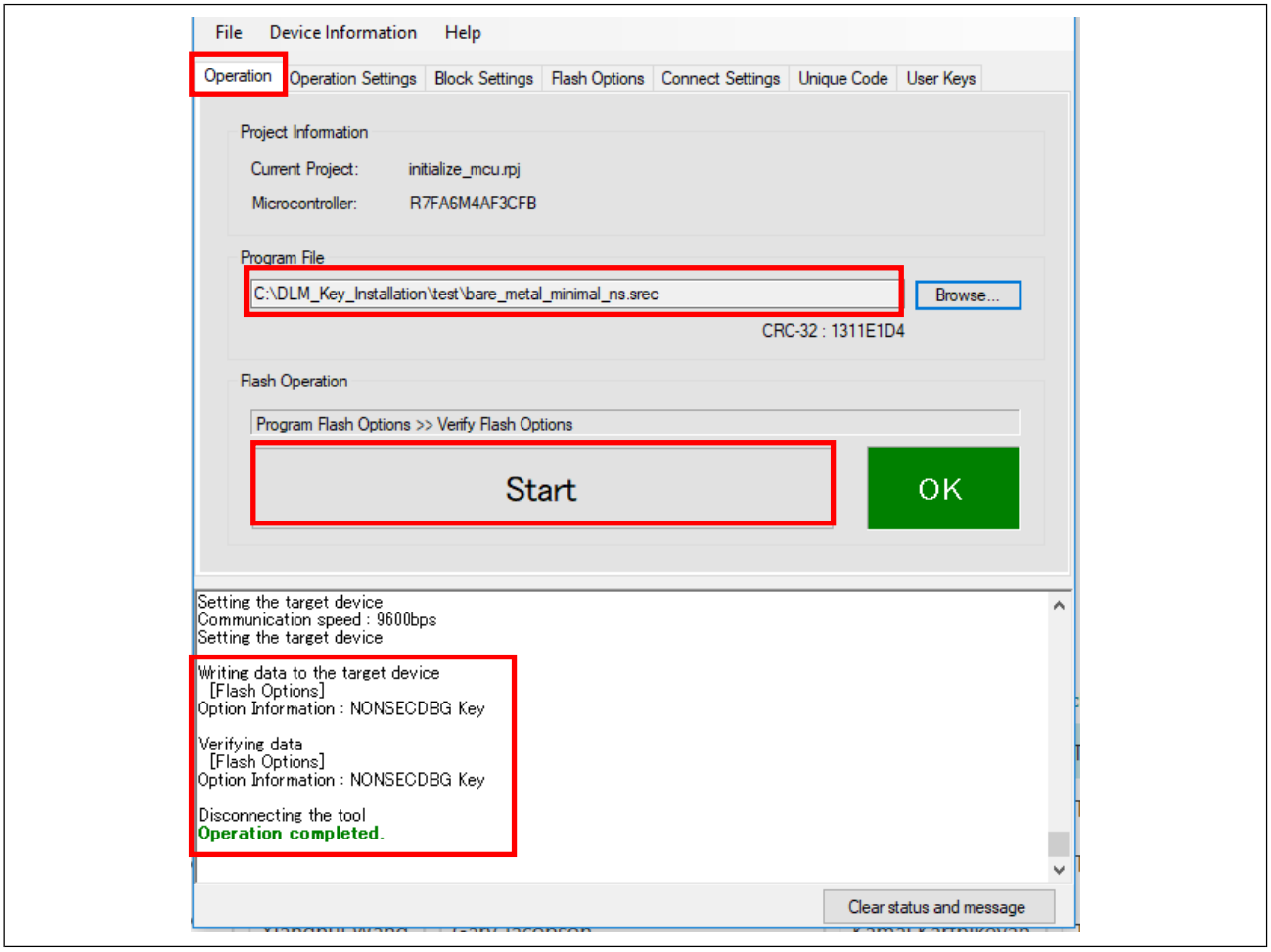

图 57. 使用 RFP 安装 NONSECDBG_KEY

3.6 需要经过身份验证的 DLM 状态转换

本节将提供需要经过身份验证的 DLM 状态转换的操作步骤。假设已经使用前面讨论的步骤安装了 SECDBG_KEY 和 NONSECDBG_KEY。

请注意，出于练习目的，用户可以在 SSD 状态下安装 RMA_KEY。但是，除非要进行产品退货，否则请不要使用 RMA_KEY 将 DLM 状态转换到 RMA_REQ 状态。一旦转换到 RMA_REQ 状态，“**All erase**”（全部擦除）操作将会失效。MCU 将被锁定，无法进行调试或使用串行编程端口进行重新编程。

此外，仅应在安全环境中转换到 RAM_REQ。

3.6.1 需要经过身份验证的非安全调试状态到安全调试状态转换

假设当前的器件生命周期状态是 NSECSD，并且之前已经安装 SECDBG_KEY。用户可以读取**器件信息**以确认这一点。

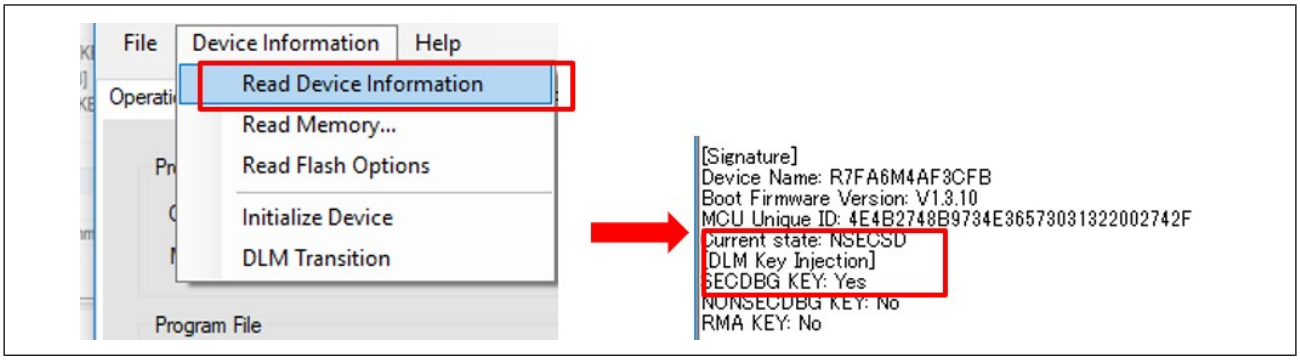

图 58. 确认 MCU 处于 NSECSD 状态并且已安装 SECDBG_KEY

使用以下步骤使器件生命周期状态退回到 SSD：

1. 从 RFP 的“Device Information”（器件信息）菜单中，选择转换到 **SSD**。

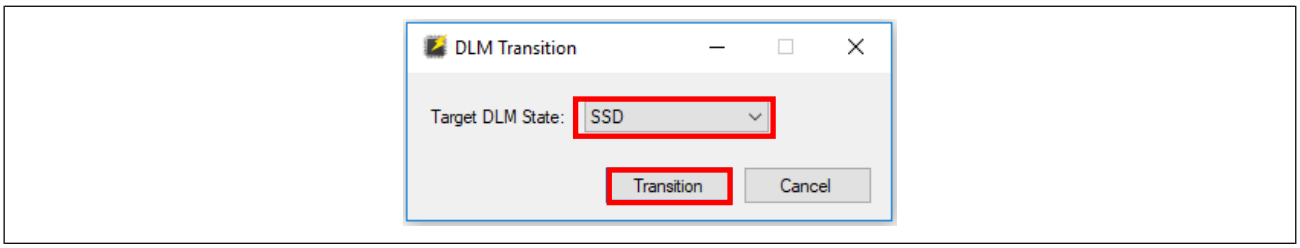

图 59. 选择使 MCU 器件生命周期状态退回到 SSD

2. 如果器件处于 NSECSD 状态，并且 MCU 上已经安装了 SECDBG 密钥，则会弹出如下提示。按照提示提供 SECDBG 密钥身份验证数据，然后单击“**OK**”（确定）。如果已经把根据图 48 生成的 `SECDBG.rkey` 安装到 MCU 中，则此数据为 000102030405060708090A0B0C0D0E0F。

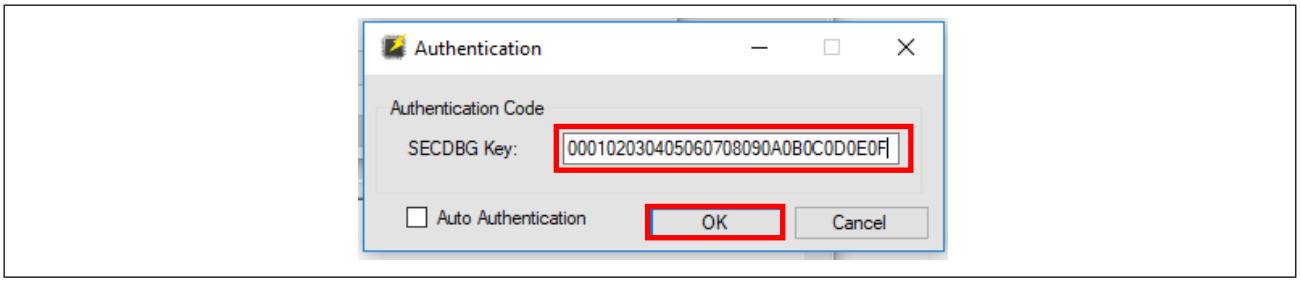

图 60. 提供 SECDBG 的身份验证密钥

3. 用户现在可以确认器件生命周期状态已转换回 SSD 状态。

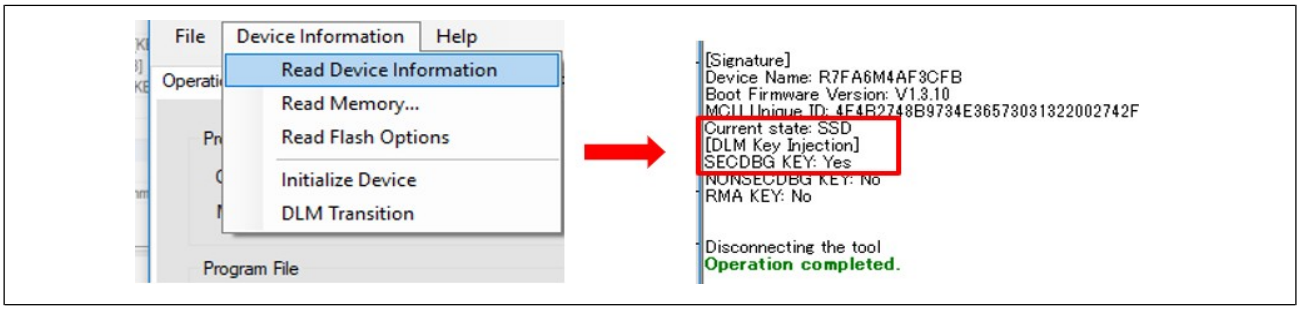

图 61. 确认器件生命周期转换回 SSD

3.6.2 需要经过身份验证的已部署状态到非安全调试状态转换

如第 2.5 节所述，如果部署状态是 DPL，则可以使 MCU 器件生命周期状态从 DPL 退回到 NSECSD。同样，用户可以通过读取**器件信息**来确认 MCU 器件生命周期状态，如下图所示。

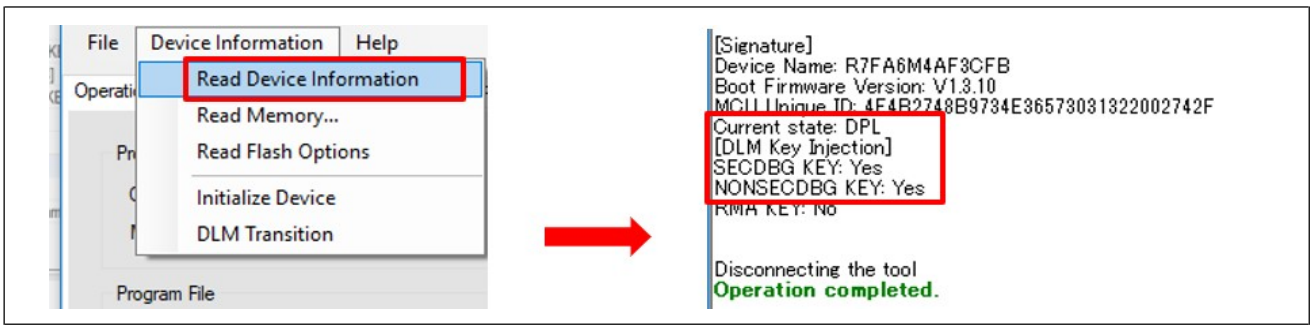

图 62. 确认器件生命周期状态和 DLM 密钥状态

通过以下步骤使器件生命周期状态退回到 NSECSD。

1. 选择转换到 NSECSD。

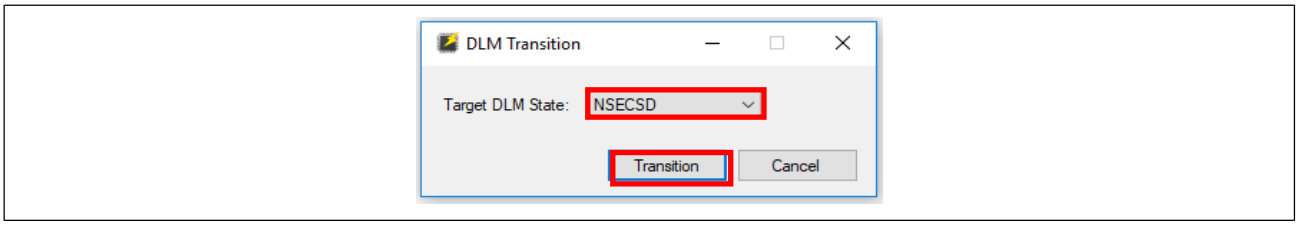

图 63. 选择转换到 NSECSD

2. 如果器件处于 DPL 状态，并且 MCU 上已经安装了 NONSECDBG_KEY，则会弹出如下提示。按照提示提供 NONSECDBG 密钥，然后单击“**OK**”（确定）。如果已经把根据图 49 生成的 `NONSECDBG.rkey` 安装到 MCU 中，则此数据为 **0101020304050607080090A0B0C0D0E0F**。

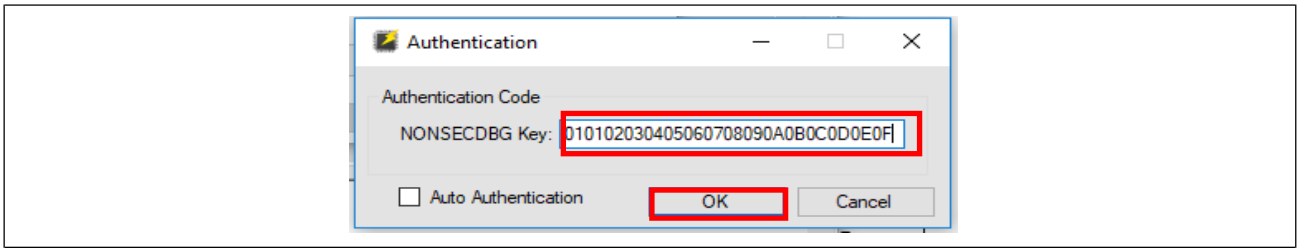

图 64.提供 NONSECDBG_KEY 的身份验证密钥

3. 用户现在可以确认器件生命周期状态已退回到 NSECSD 状态。

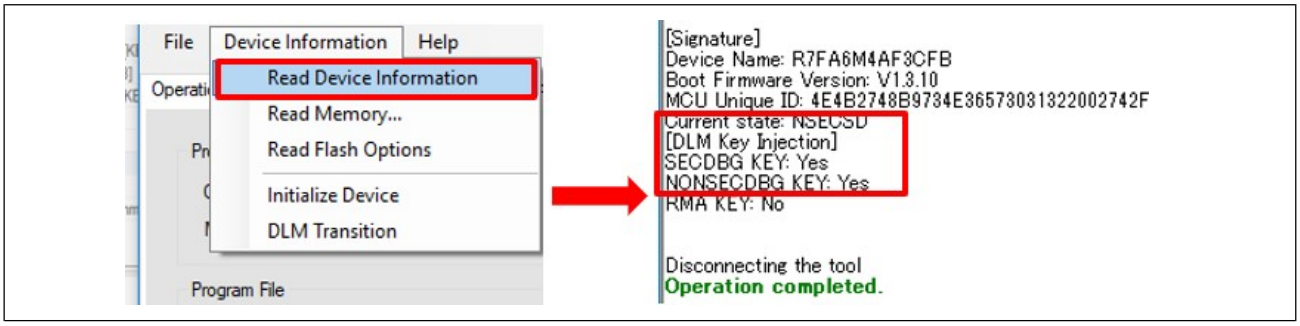

图 65. 确认器件生命周期已转换回 NSECSD

4. 参考资料

可通过 renesas.com 获取：

- 《瑞萨 RA6M4 系列用户手册：硬件》
- 《灵活配置软件包 (FSP) 用户手册》
- 《利用瑞萨安全 MPU 保护静态数据》

5. 附录

5.1 术语表

术语	含义
SCE9	安全加密引擎 9 是瑞萨 Arm® Cortex®-M33 MCU 上的一个硬件单元
器件证书	用于标识单个器件的唯一证书，该证书带有数字签名，可确定证书来自已知来源且未遭到篡改，并且器件是可信的。
信任根	信任根是高度可靠的硬件、固件和软件组件，用于执行特定关键安全功能。(https://csrc.nist.gov/projects/hardware-roots-of-trust)
SCE	安全加密引擎 – MCU 中的一个模块，可实现高效、低功耗的加密加速、TRNG（真随机数生成）和加密密钥的创建和隔离。
PKI	公钥架构 – 创建、管理、分发、使用、存储和撤销数字证书所需的一组角色、策略和程序，通常用于通过公钥加密管理安全身份。
密钥对	成对生成的非对称密钥，包括公钥和私钥。私钥仅由一方秘密持有，可用于确定该方的身份。公钥可自由分发，并与私钥具有唯一关联。
安全代码	位于内部闪存安全区域中的一个函数或一组函数，由 MPU 定义和执行。这些安全功能可以访问安全数据和非安全数据区域。
非安全代码	位于内部闪存非安全区域中的一个函数或一组函数。这些非安全代码无法访问安全区域。它们只能访问非安全区域。
HUK	硬件唯一密钥。这是存储在 RA 产品家族 MCU 中的唯一密钥。
质询字符串	在主机应用程序中随机生成的字符串。主机应用程序使用该字符串 来验证目标对私钥的所有权。
唯一 ID	每个 RA 产品家族 MCU 的唯一标识值，存储在 MCU 内。SCE 在封装密钥时使用该唯一 ID。
质询响应字符串	对质询字符串的响应。质询响应字符串是质询数据的签名，是通过使用接收者的私钥对质询字符串进行签名而创建的。

网站和支持

如需了解 RA 系列的关键元素、下载组件和相关文档以及获得支持，请访问以下虚拟 URL。

EK-RA6M4 资源	renesas.com/ra/ek-ra6m4
RA 产品信息	renesas.com/ra
RA 产品支持论坛	renesas.com/ra/forum
RA 灵活配置软件包	renesas.com/FSP
瑞萨支持	renesas.com/support

版本历史记录

版本	日期	说明	
		页码	摘要
1.00	2020 年 10 月 1 日	—	首次发行版本文件
1.10	2020 年 12 月 9 日	—	更新了将瑞萨公钥导入 Kleopatra 的过程
1.11	2021 年 4 月 30 日	—	改善措辞。

注意

1. 本文档中电路、软件和其他相关信息的描述仅用于说明半导体产品的操作和应用示例。用户应对产品或系统设计中电路、软件和信息的纳入或任何其他用途承担全部责任。对于您或第三方因使用这些电路、软件或信息而引起的任何损失和损害，Renesas Electronics 不承担任何责任。
2. Renesas Electronics 特此声明，对于因使用本文档中所述的 Renesas Electronics 产品或技术信息（包括但不限于产品数据、图纸、图表、程序、算法和应用示例）而引起的侵权或与第三方有关的专利、版权或其他知识产权的任何其他索赔，概不承担任何责任和赔偿。
3. 对 Renesas Electronics 或其他公司的任何专利、版权或其他知识产权均不授予任何明示、暗示或其他形式的许可。
4. 您应负责确定需要从任何第三方获得哪些许可，并在需要时为合法进口、出口、制造、销售、使用、分销或以其他方式处置包含 Renesas Electronics 产品的任何产品获得此类许可。
5. 不得对 Renesas Electronics 产品的全部或部分进行更改、修改、复制或逆向工程。对于因更改、修改、复制或逆向工程而导致您或第三方蒙受的任何损失或损害，Renesas Electronics 不承担任何责任。
6. Renesas Electronics 产品根据以下两个质量等级进行分类：“标准”和“优质”。Renesas Electronics 每种产品的预期应用取决于产品的质量等级，具体如下所示。
   “标准”：计算机、办公设备、通信设备、测试和测量设备、视听设备、家用电器、机械工具、个人电子设备、工业机器人等
   “优质”：运输设备（汽车、火车、轮船等）；交通管制（交通信号灯）；大型通信设备；关键金融终端系统；安全控制设备等
   除非在 Renesas Electronics 数据手册或 Renesas Electronics 其他文档中明确指定为高可靠性产品或用于恶劣环境的产品，否则 Renesas Electronics 产品不适合或不授权用于可能对人类生命构成直接威胁或造成人身伤害（人造生命支持设备或系统；手术植入物等），或者可能造成严重的财产损失（空间系统、海底中继器、核动力控制系统、飞机控制系统、关键设备系统、军事装备等）的产品或系统。对于因使用任何与 Renesas Electronics 数据手册、用户手册或其他 Renesas Electronics 文档不一致的 Renesas Electronics 产品而引起的您或任何第三方所造成的任何损坏或损失，Renesas Electronics 不承担任何责任。
7. 没有任何半导体产品是绝对安全的。尽管 Renesas Electronics 的硬件或软件产品中可能实施了任何安全措施或功能， Renesas Electronics 对因任何漏洞或侵骇（包括但不限于以任何未经授权的方式访问或使用 Renesas Electronics 产品或使用 Renesas Electronics 产品的系统）而产生的任何后果概不负责。RENESAS ELECTRONICS 不担保或保证 RENESAS ELECTRONICS 产品或使用 RENESAS ELECTRONICS 产品创建的任何系统不会被破坏，或者可免于数据损坏、攻击、病毒、干扰、黑客攻击、数据丢失或失窃或其他安全入侵（“漏洞问题”）。RENESAS ELECTRONICS 不承担由任何漏洞问题引起的或与之相关的任何和所有责任或义务。此外，在适用法律允许的范围内，RENESAS ELECTRONICS 不对本文件和任何相关或附带的软件或硬件提供任何和所有明示或暗示的保证，包括但不限于对适销性或对特定用途的适用性的暗示保证。
8. 使用 Renesas Electronics 产品时，请参见最新的产品信息（数据手册、用户手册、应用笔记、可靠性手册中的“处理和使用半导体器件的一般说明”等），并确保使用条件符合 Renesas Electronics 在最大额定值、工作电源电压范围、散热特性和安装等方面的规定。对于因在超出上述规定范围的条件下使用 Renesas Electronics 产品而引起的任何失常、故障或事故，Renesas Electronics 不承担任何责任。
9. 尽管 Renesas Electronics 致力于提高 Renesas Electronics 产品的质量和可靠性，但半导体产品具有特定的特性，例如在特定速率下发生故障以及在某些使用条件下出现故障。除非在 Renesas Electronics 数据手册或 Renesas Electronics 其他文档中指定为高可靠性产品或用于恶劣环境的产品，否则 Renesas Electronics 的产品将不受抗辐射设计的约束。用户应负责采取安全措施，以防止人身伤害、火灾造成的伤害，和/或因 Renesas Electronics 产品发生故障或失常而对公众造成的危险，例如硬件和设备的安全设计，包括但不限于冗余、火控和故障预防、针对老化退化的适当处理或任何其他适当的措施。由于对微型计算机软件进行评估非常困难且无实操性，因此用户有责任评估自己生产的最终产品或系统的安全性。
10. 请联系 Renesas Electronics 销售办事处，以获取有关环境事宜的详细信息，例如每个 Renesas Electronics 产品的环境相容性。用户有责任认真、充分地研究有关纳入或使用受控物质的适用法律和法规（包括但不限于欧盟 RoHS 指令），并按照所有适用法律和法规使用 Renesas Electronics 产品。对于因您未遵守适用的法律和法规而造成的损坏或损失，Renesas Electronics 不承担任何责任。
11. Renesas Electronics 产品和技术不得被用于或纳入为任何适用的本国或外国法律、法规所禁止制造、使用或销售的产品或系统范围内。用户应遵守由对当事方或交易拥有管辖权的任何国家/地区的政府颁布和管理的任何可适用的出口控制法律和法规。
12. 应由Renesas Electronics 产品的购买方或分销商，或者对产品进行分发、处置或以其他方式出售或转让给第三方的任何其他当事方，负责将本文档中阐明的内容和条件提前通知前述第三方。
13. 未经 Renesas Electronics 事先书面同意，不得以任何形式全部或部分重印、再现或复制本文档。
14. 如果对本文档中包含的信息或 Renesas Electronics 产品有任何疑问，请联系 Renesas Electronics 销售办事处。

（注 1）本文档中的“Renesas Electronics”是指 Renesas Electronics Corporation，也包括其直接或间接控制的子公司。
（注 2）“Renesas Electronics 产品”是指 Renesas Electronics 开发或制造的任意产品。

（版本 5.0-1 2020 年 10 月）

公司总部

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

联系信息

有关产品、技术、文档最新版本或离您最近的销售办事处的更多信息，请访问：www.renesas.com/contact/。

商标

Renesas 和 Renesas 徽标是 Renesas Electronics Corporation 的商标。所有商标和注册商标都是各自所有者的财产。