DA 证书链之初体验

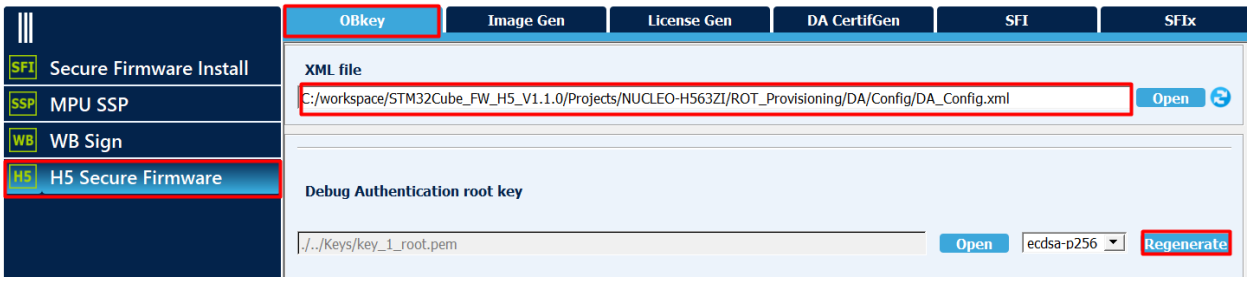
关键字：DA， Debug Authentication, 调试认证, DA 证书, 证书链

1. 前言

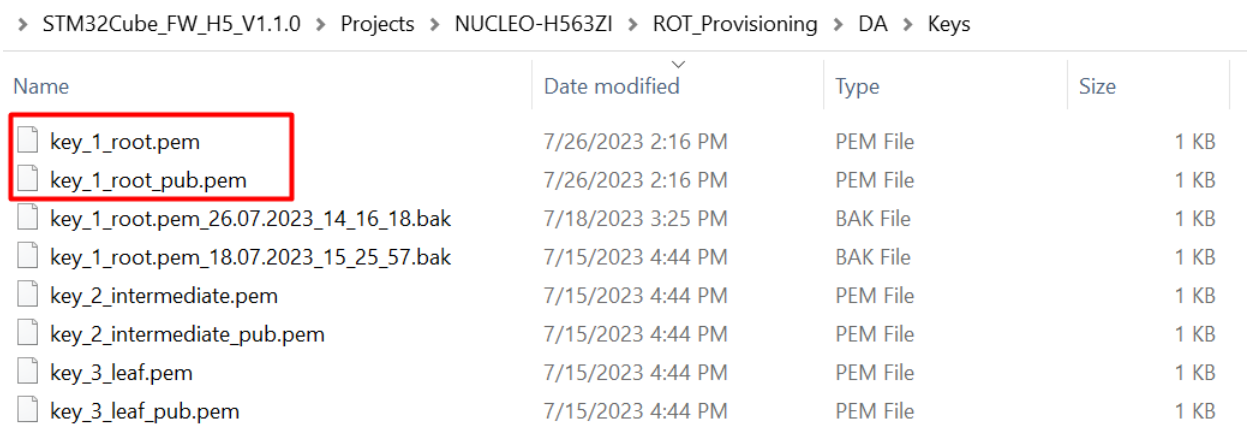
本文是上文 《STM32H5 DA 之初体验(带 TrustZone)》的后续之作。倘若你还没有阅读此文，那么建议你先阅读下，然后再回过头来阅读本文。

之前我们已经讲过了如何通过 DA 认证来回退芯片产品状态，或者重新打开调试口，这样开发人员在芯片为 Closed 状态下时仍可以调试芯片。在这个 DA 认证过程中，有使用到两个东西：证书和私钥，它与之前已经预配置到芯片内部的 obk 文件是对应的关系。也就是说, 如果你已经预配置了芯片, 但对应的私钥文件或者证书丢失或忘记保存了, 那么此芯片多半是无法再还原了, 除非你找到对应的私钥和证书.

私钥是如何来的？如上文所述, 是通过 TPC 工具生成的, 如下所示：

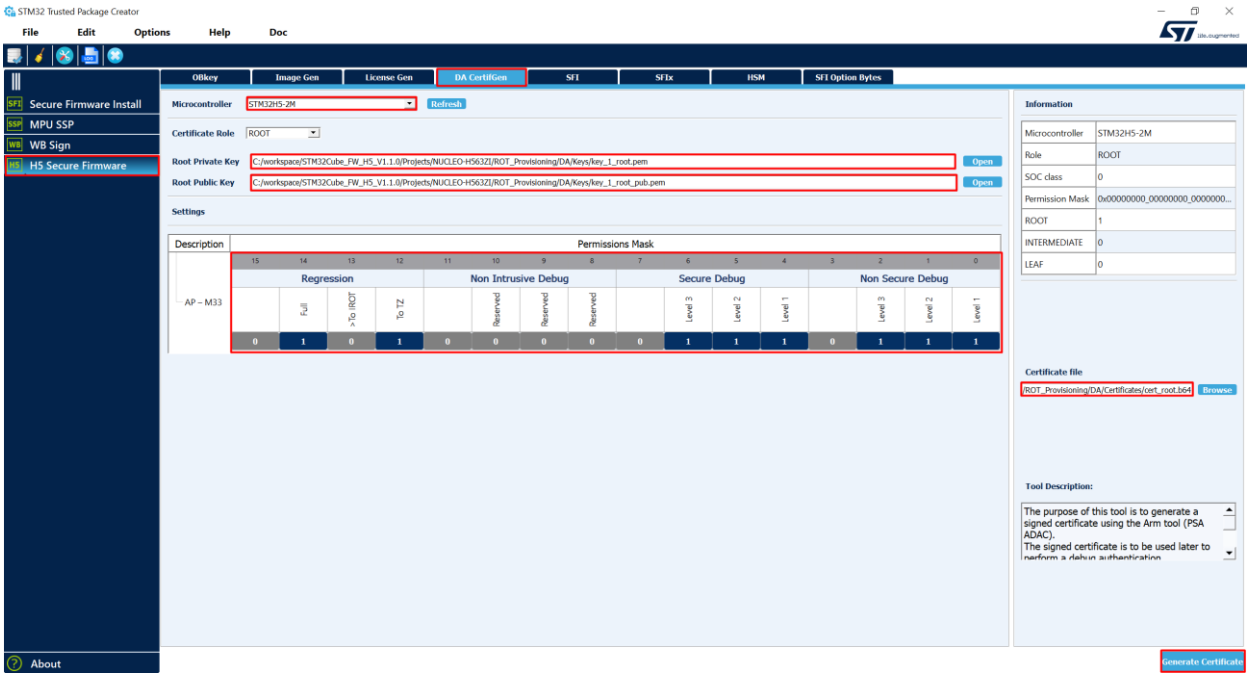


点击上图中的 Regenerate 按键将自动生成公钥私钥对.



当然你也可以使用其它工具来生成, 比如 openssl 工具, 只要是 ecdsa-p256 类型的密钥即可.

证书是怎么来的？也是通过 TPC 工具生成的 。



如上图所示, 输入根密钥的私钥和公钥, 证书类型选择为 “ROOT”, 然后在操作许可内选择所有权限(完全回退/半回退 + 安全调试 + 非安全调试).最终点击 “Generate Certificate” 按键, 生成根证书. 由于这里选择的类型为 ROOT, 输入的密钥也为根密钥. 所以最终生成的证书为根证书. 它所许可的权限是最高的.

对于一个产品的开发, 可能由不同团队一起合作开发的, 比如, 有一团队, 专门负责开发 secure 部分的代码, 向 non secure 工程提供 API 接口, 将那些关键算法放入到 secure 世界加以保护, 仅仅面向 non secure 工程开放调用接口. 而另一个开发团队, 他们仅仅开发 non secure 部分的代码, 实现应用层大部分功能, 需要时调用 secure 部分代码开放出来的接口以实现特定的功能. 那么对于这种不同的开发团队, 其所允许的权限必须有所不同, 开发 secure 部分的团队, 必然希望他们自己能调试 secure 部分的代码, 但不希望开发 non secure 部分的开发团队也可以调试 secure 代码. 但又要允许不影响他们可以调试自己 non secure 部分的代码, 且可随意回退 non secure 部分代码. 如此复杂的权限控制, 就需要多个证书来实现了. 这就是证书链的意义所在.

2. 证书链介绍

在产品的开发过程中, 我们假设有三个团队:

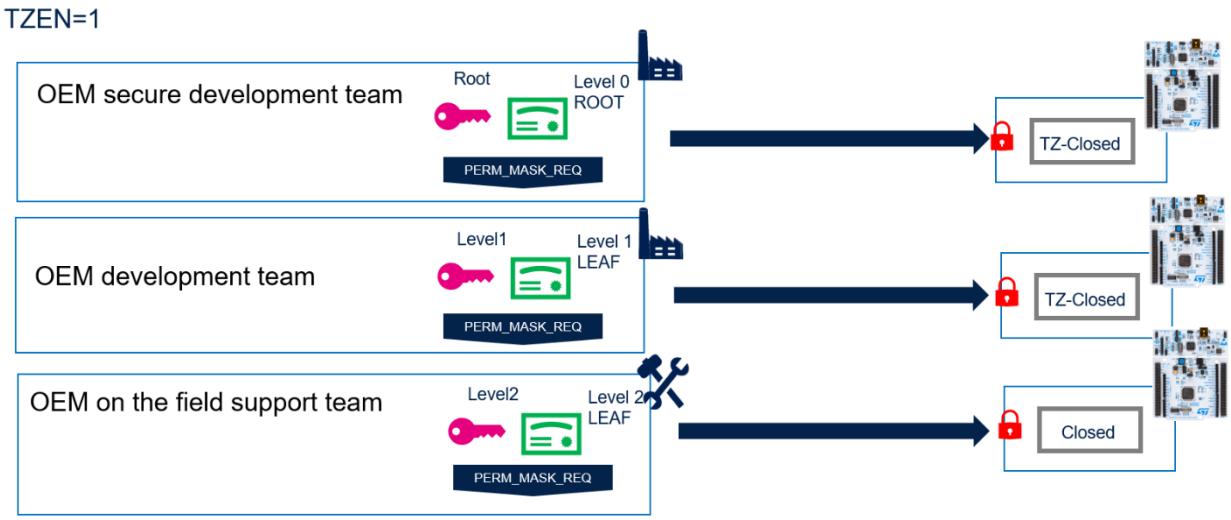
- 安全开发团队: 负责开发 secure 部分代码, 并向 non secure 提供 API 接口. 当芯片烧录完 secure 代码后, 芯片会被设置为 TZ_Closed 状态.
- OEM 开发团队: 负责开发 non secure 部分代码, 无法直接访问 secure 世界的代码, 但能调用 secure 提供的特定的 API 接口, 以完成特定功能. 此团队拿到芯片时, 芯片已经处于 TZ_Closed 状态. 在此状态下, 开发 non secure 工程不受任何限制. 可以随意调试 NS 工程.
- 现场技术支持团队: 产品到达终端客户, 若出现任何问题, 需要现场进行技术支持.

对于这三个团队, 需要授予不同的权限. 安全开发团队需要授予平台软硬件完全访问的权限. 且安全开发团队可向 OEM 团队授予权限. 完全访问权限就包括了可完全回退+部分回退, S+NS 调试.

OEM 开发团队的权限应该仅仅局限在 NS 部分, 不应该影响其 NS 代码的调试, 当然回退也不应该限制. 仅需要限制其调试 S 代码的权利. 因此, OEM 团队包括的权限应该有: 完全回退+部分回退, NS 调试. 除此之外, OEM 开发团队应可向现场技术支持团队授予权限的能力.

现场技术支持团队的权限应该仅限于回退的权利.而且仅能完全回退. 以便检查硬件方面的问题.

针对这三个团队的授权情况, 引入三级证书. 根证书, 一级证书和二级证书. 分别对应这三个团队. 这种操作授权就放入到证书里边.



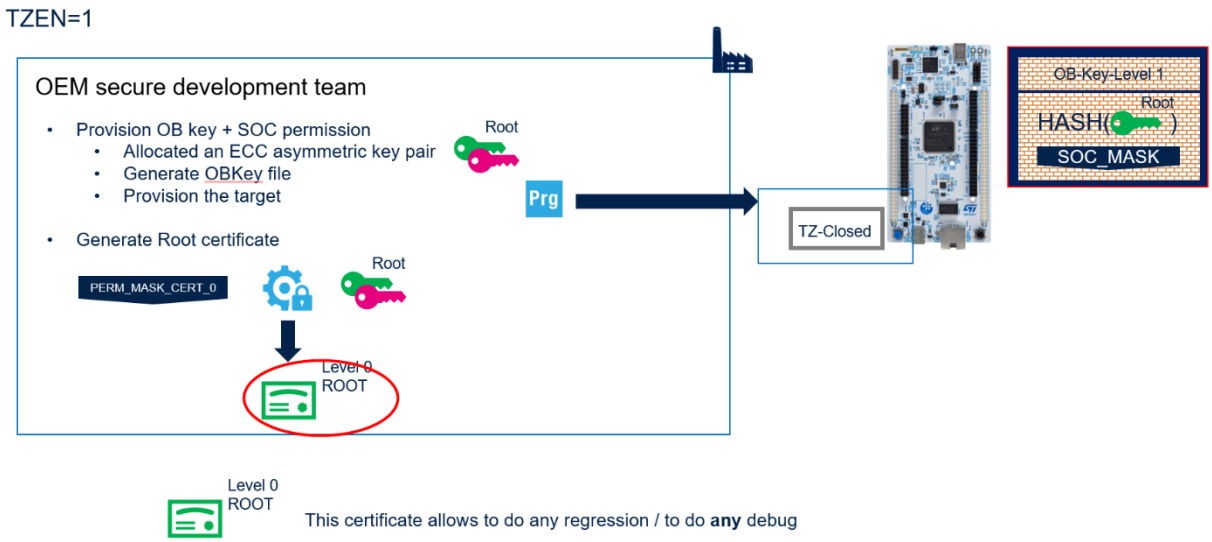
如上图所示, 安全开发团队使用自己的根证书和根密钥. OEM 开发团队使用自己的一级密钥和一级证书; 而现场技术支持团队则使用自己的二级密钥和二级证书. 各自有自己的一套密钥和证书. 这里的密钥是指私钥. 那么这些密钥和证书又是如何来的呢?

2.1. 密钥和证书

这里的密钥其实是指私钥, 它和公钥是一对的. 也就是说, 三个团队其实都有自己的密钥对. 在做 DA 时, 会使用到私钥和证书. 首先我们来看看这个密钥对是如何来的呢?

如前文第一章节的前言部分, 我们就介绍了根密钥对的来源, 它是由 TPC 工具内在不同的路径下点击 Regenerate 按键后生成的. 那么对于 OEM 开发团队和现场技术支持团队的密钥对, 也是这么生成的. 当然, 也可以通过第三方工具如 openssl 这类的工具生成. 我们将三个团队的密钥(指私钥)分别叫做根密钥, 一级密钥, 二级密钥. 它们是三个相互独立的密钥.

2.1.1. 根证书

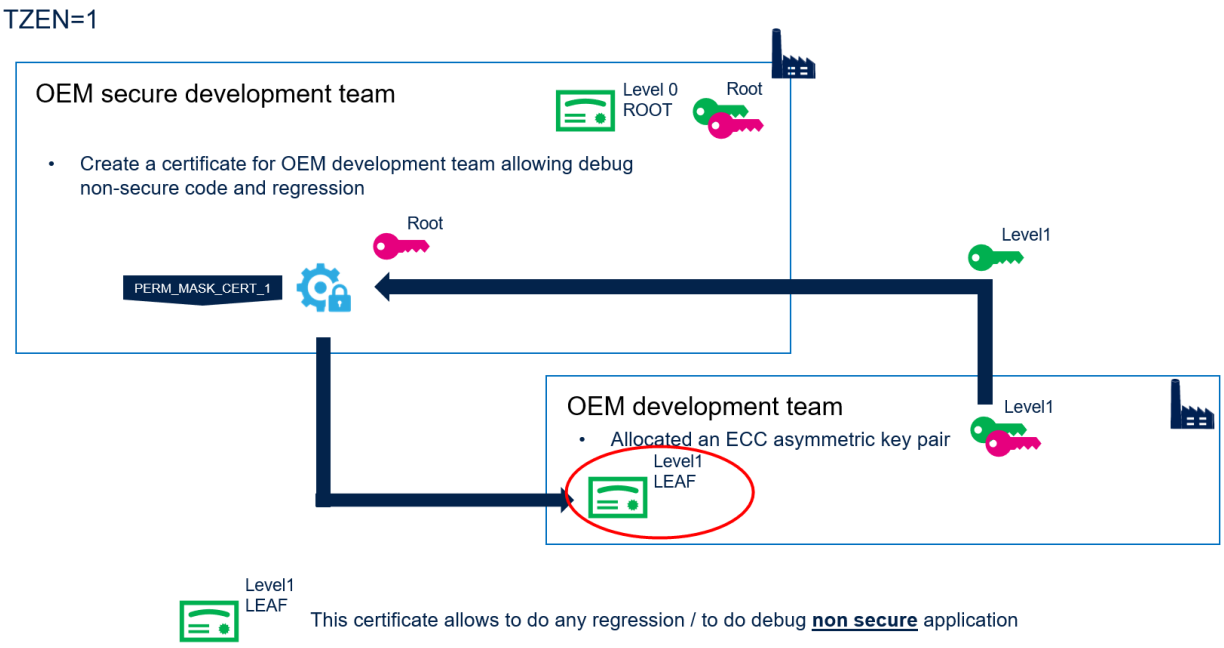


如上图所示, 根证书是安全开发团队自己用私钥对自己的公钥以及操作许可签名产生的. 这个证书授权完全回退, 以及 S 调试+NS 调试.

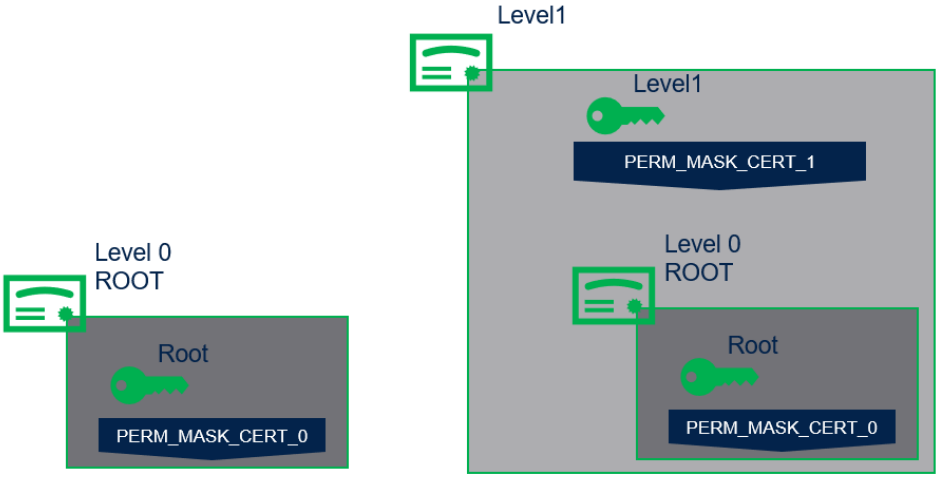
它的生成正如前文第 1 章节如所描述, 生成的就是根证书.

2.1.2. 一级证书

一级证书是由安全开发团队负责生成, 但使用方却是 OEM 开发团队. 也就是说, 它是由安全开发团队授权给 OEM 开发团队的证书.

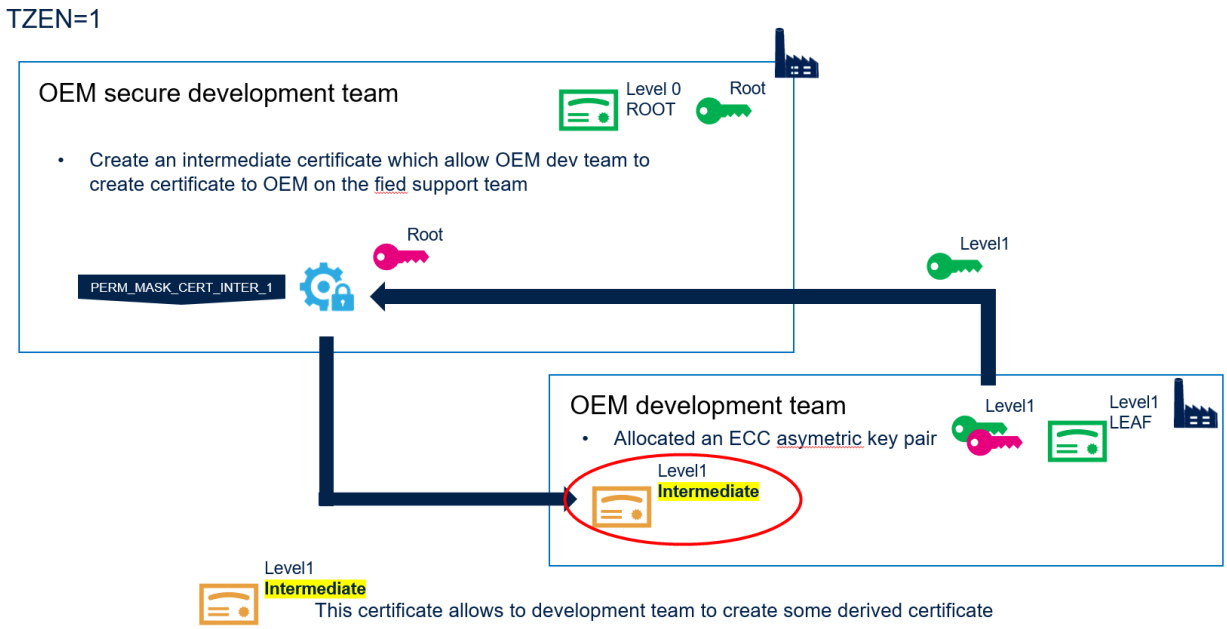


如上图所示, 它是拿 OEM 开发团队的公钥出来, 用安全开发团队的私钥进行签名所产生, 再加上授权许可. 它是安全开发团队的根证书的子证书:



2.1.3. 中间证书

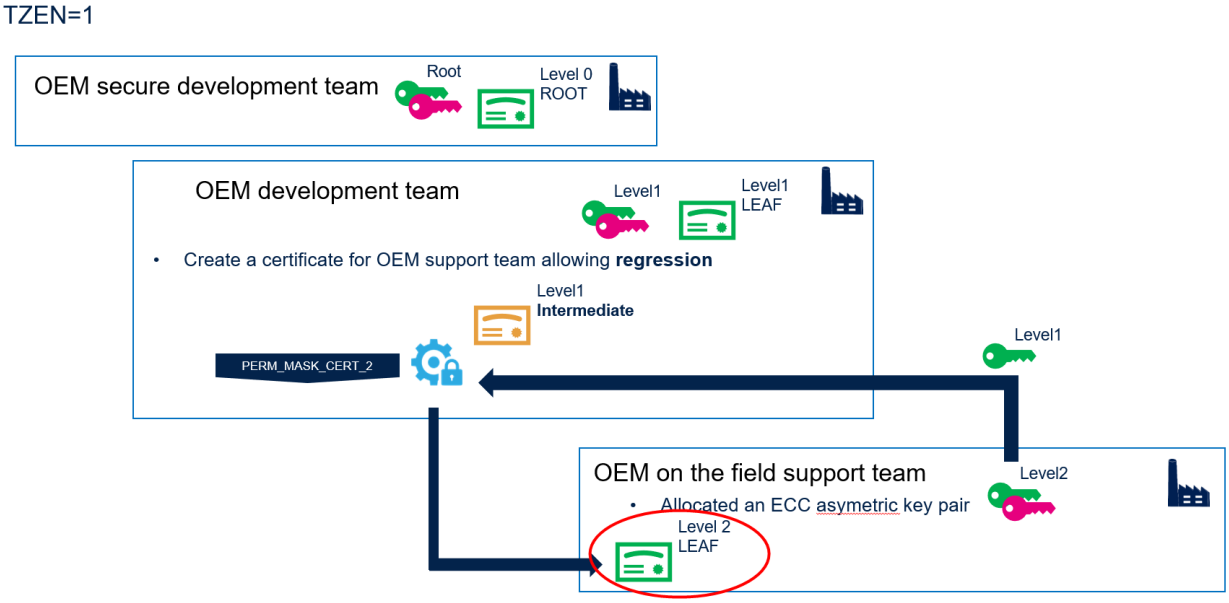
中间证书也是一级证书, 它也是由安全团队的根证书的子证书, 由安全开发团队授权而来. 与一级证书不同地是, 它只能用来派生二级证书, 并不能直接拿来用, 比如 DA 回退, DA 调试. 它仅仅用来颁发二级证书, 给证书管理者用的.



如上图所示, 中间证书是拿 OEM 开发团队的公钥, 用安全开发团队的私钥签名产生, 再加上操作许可信息. 它是给 OEM 开发团队用的, 仅仅用于派生二级证书. 它也是安全开发团队的根证书的子证书.

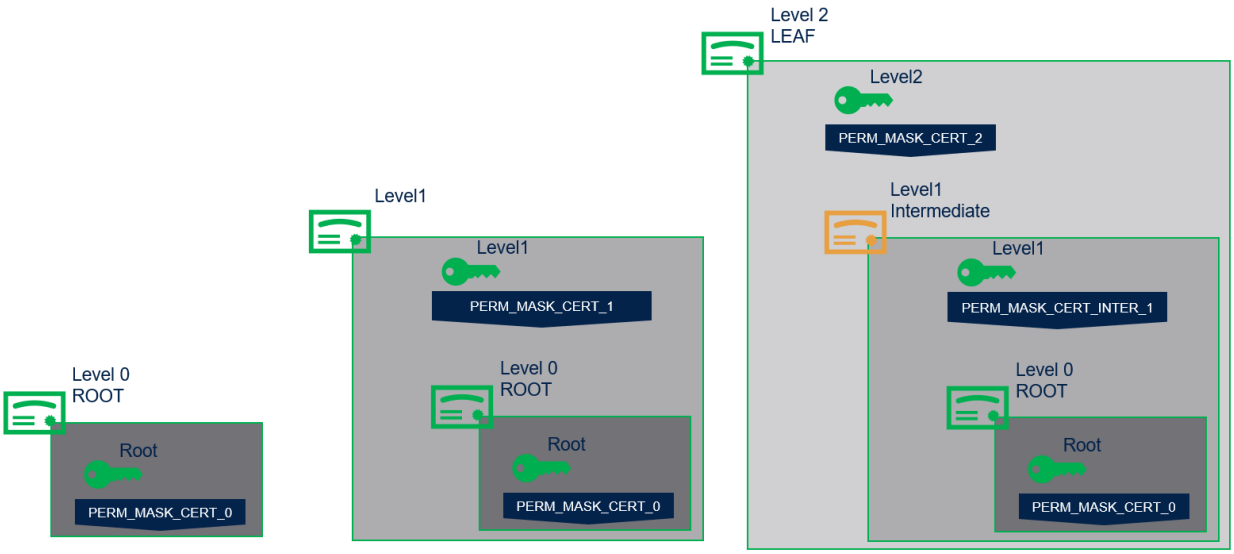
2.1.4. 二级证书

二级证书是给现场技术支持团队用的. 它是由中间证书派生而来:



如上图所示, 在从中间证书派生二级证书时, 拿二级公钥出来, 用一级私钥进行签名的, 再加上操作许可信息. 二级证书生成过程完全是由 OEM 开发团队负责的. 这里需要注意地是, 二级证书是由中间证书派生而来, 并不是由一级证书派生.

STM32H5 支持这三级证书: 根证书, 一级证书, 中间证书和二级证书, 其相互关系如下图所示:

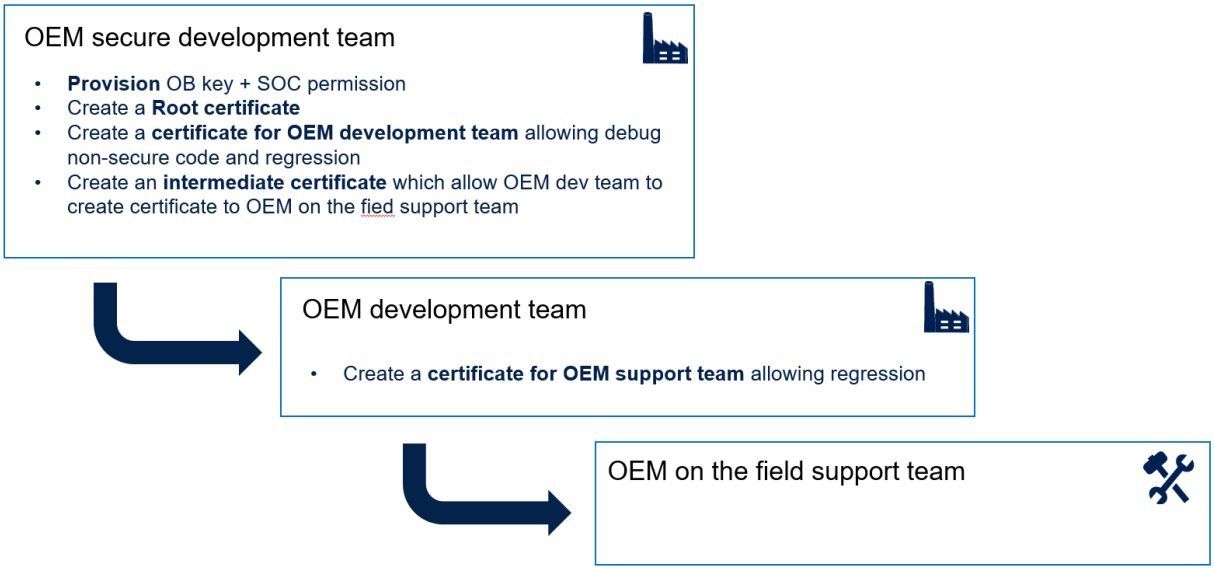


3. 动手实验

3.1. 实验概览

本实验将分别扮演安全开发团队, OEM 开发团队和现场技术开发团队, 分别生成根证书, 一级证书, 中间证书和二级证书, 并验证此证书的有效性.

TZEN=1



3.2. 安全开发团队

安全开发团队有以下几个任务 :

- 预配置 OBKey+芯片操作许可.
- 生成根证书
- 为 OEM 开发团队生成一级证书
- 为 OEM 开发团队生成中间证书

3.2.1. 预配置 OBKey+芯片操作许可

此过程与《STM32H5 DA 之初体验(带 TrustZone)》的 3.2 节+3.4 节完全一致。这里不再赘述。

3.2.2. 生成根证书

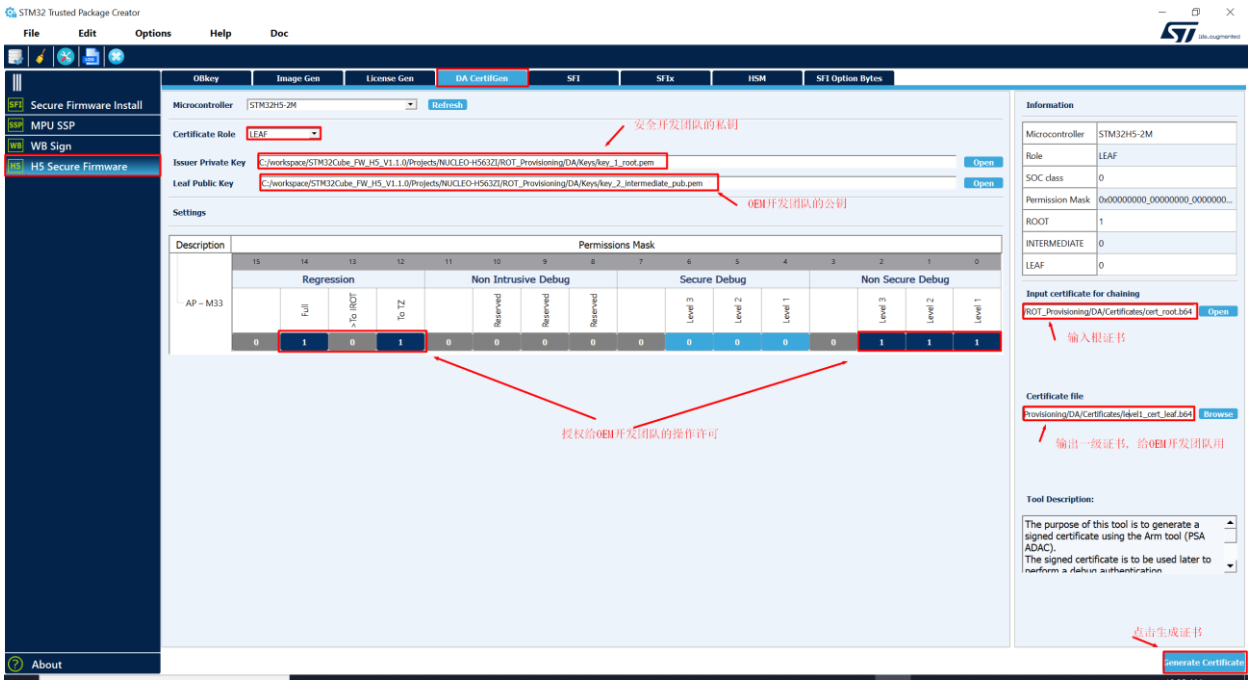
此过程与《STM32H5 DA 之初体验(带 TrustZone)》的 3.3 节完全一致。这里不再赘述。

3.2.3. 测试根证书的有效性

此过程与《STM32H5 DA 之初体验(带 TrustZone)》的第 4 章完全一致。这里不再赘述。

3.2.4. 为 OEM 开发团队生成一级证书

为 OEM 开发团队生成一级证书首先得拿到 OEM 开发团队的公钥. 关于 OEM 开发团队如何生成自己的公钥私钥对, 请参考 3.3.1 节.



如上 图所示, 在 Certificate Role 处选择 LEAF, 在 Issuer Private Key 处输入根密钥: C:/workspace/STM32Cube_FW_H5_V1.1.0/Projects/NUCLEO-H563ZI/ROT_Provisioning/DA/Keys/key_1_root.pem, 用它来给证书签名.

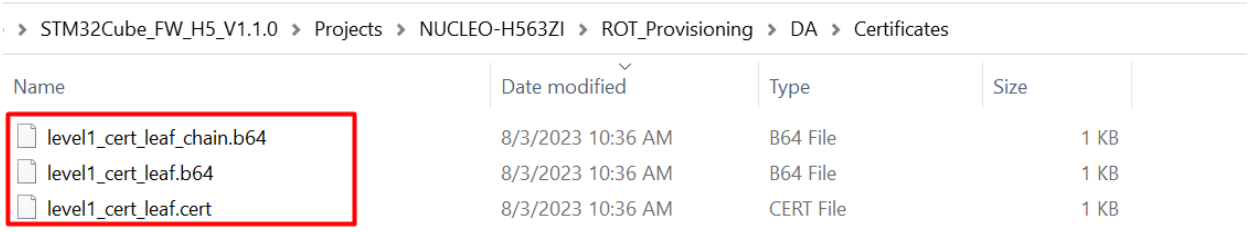
在 Leaf Public Key 处输入 OEM 开发团队的一级公钥: C:/workspace/STM32Cube_FW_H5_V1.1.0/Projects/NUCLEO-H563ZI/ROT_Provisioning/DA/Keys/key_2_intermediate_pub.pem

在 Settings 下设置开放给 OEM 开发团队的操作许可, 因为 OEM 开发团队仅仅形式发 NS 工程, 所以这里开放完全回退+半回退, +NS Debug 权限.

然后在右侧 Input certificate for chaining 处输入根证书: C:/workspace/STM32Cube_FW_H5_V1.1.0/Projects/NUCLEO-H563ZI/ROT_Provisioning/DA/Certificates/cert_root.b64

在 Certificate file 出输入需要生成的证书路径及文件名: C:/workspace/STM32Cube_FW_H5_V1.1.0/Projects/NUCLEO-H563ZI/ROT_Provisioning/DA/Certificates/level1_cert_leaf.b64

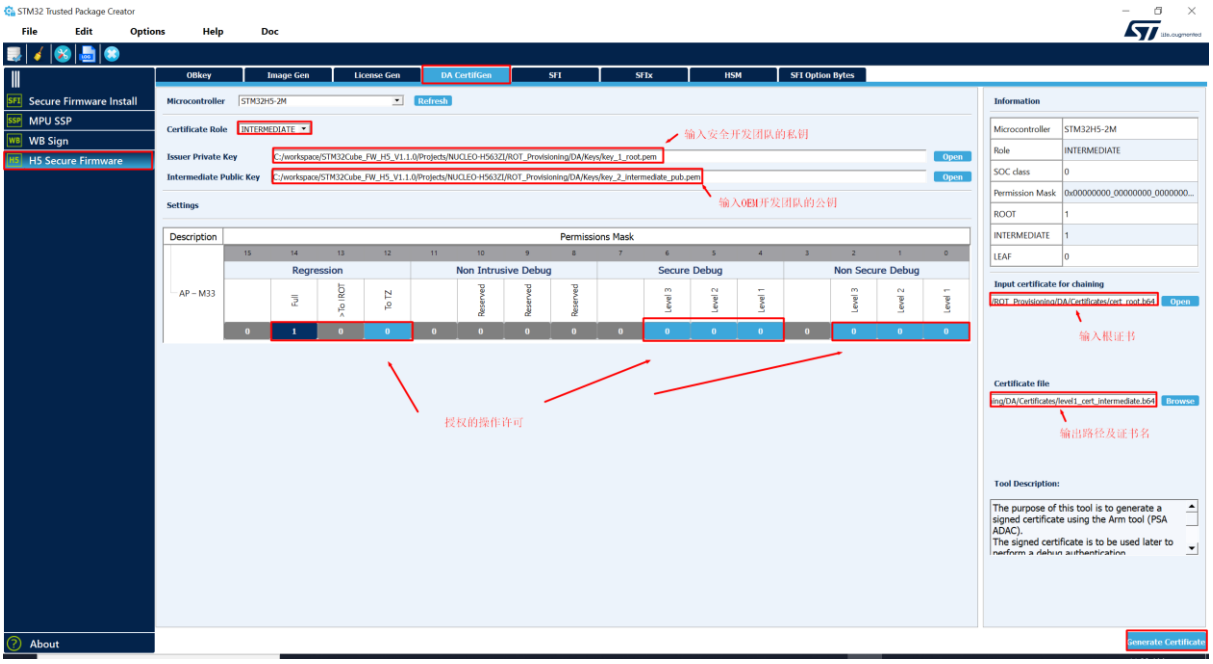
最后点击 Generate Certificate 按键生成一级证书:



其中 **level1_cert_leaf_chain.b64** 为一级证书文件, 可发给 OEM 开发团队用作 NS 工程调试和完全回退+半回退.

3.2.5. 为 OEM 开发团队生成中间证书

安全开发团队还需要给 OEM 开发团队生成中间证书, 此证书是专门用来给下一级授权的, 即生成二级证书用的.



如上图所示, 在 Certificate Role 处选择 INTERMEDIATE, 在 Issuer Private Key 处, 输出根密钥 : C:/workspace/STM32Cube_FW_H5_V1.1.0/Projects/NUCLEO-H563ZI/ROT_Provisioning/DA/Keys/key_1_root.pem, 用它给证书签名. 在 Intermediate Public Key 处, 输入 OEM 开发团队的公钥 : C:/workspace/STM32Cube_FW_H5_V1.1.0/Projects/NUCLEO-H563ZI/ROT_Provisioning/DA/Keys/key_2_intermediate_pub.pem

在下方 Settings 处, 我们仅仅选择 Full, 即仅仅开放完全回退权限. 这是由于此中间证书只是给 OEM 开发团队生成下一级证书, 即给现场技术支持团队用的, 它仅仅需要完全回退权限即可.

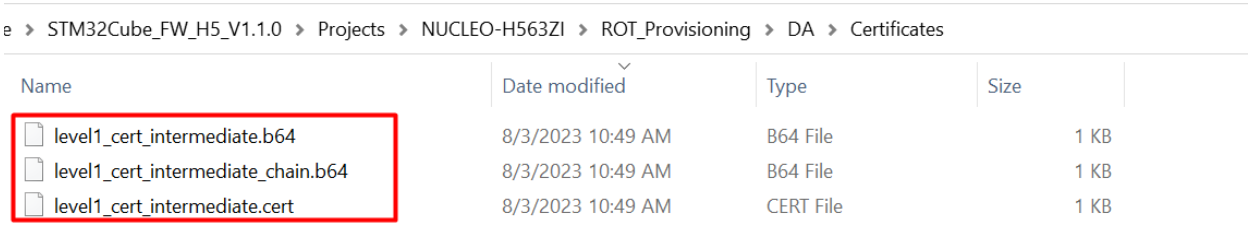
在右侧 Input certificate for chaining 处输入根证书：

C:/workspace/STM32Cube_FW_H5_V1.1.0/Projects/NUCLEO-H563ZI/ROT_Provisioning/DA/Certificates/cert_root.b64

在 Certificate file 处输入生成的证书的路径及文件名：

C:/workspace/STM32Cube_FW_H5_V1.1.0/Projects/NUCLEO-H563ZI/ROT_Provisioning/DA/Certificates/level1_cert_intermediate.b64

最后点击 Generate Certificate 按键生成中间证书：

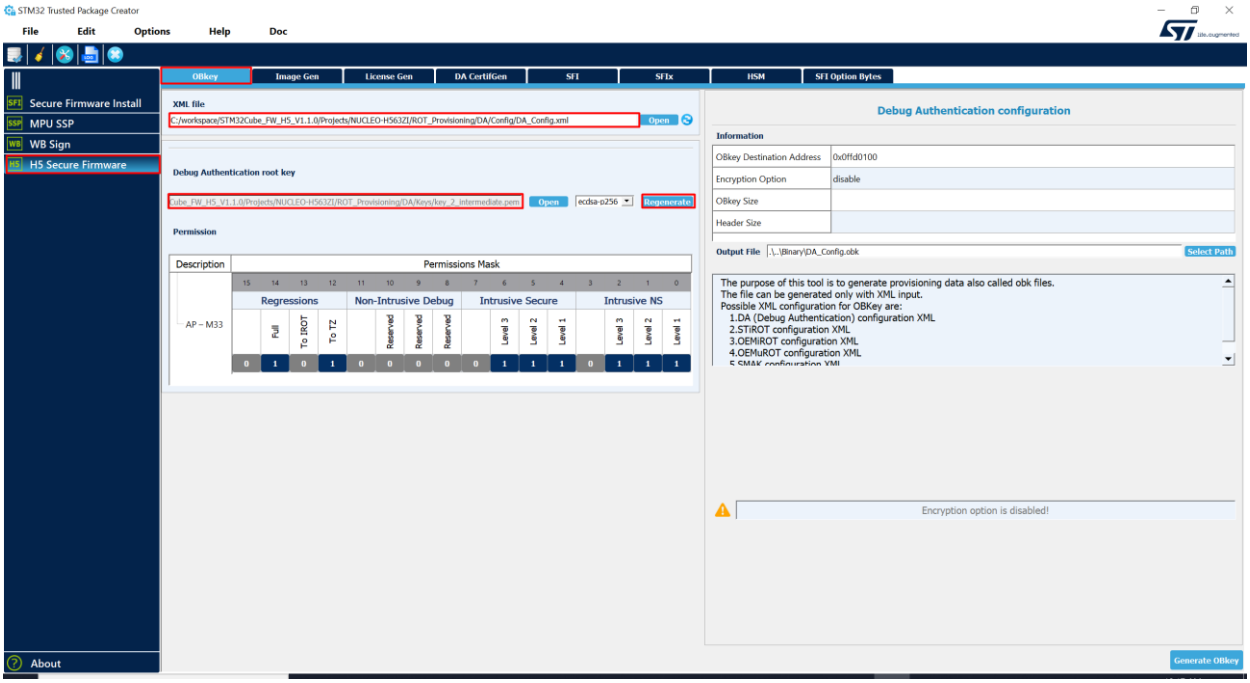


其中 **level1_cert_intermediate_chain.b64** 可发给 OEM 开发团队. 给他们生成下级证书用.

3.3. OEM 开发团队

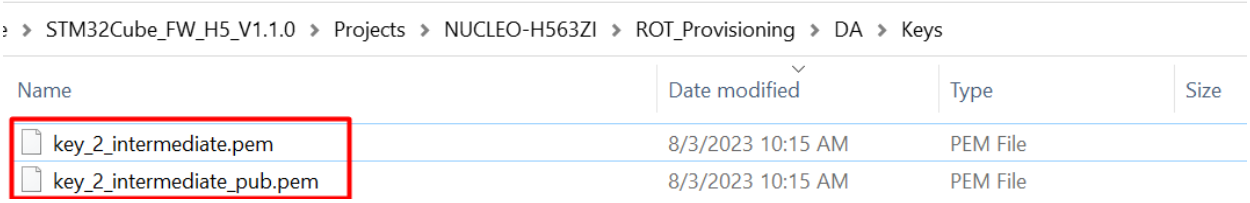
OEM 开发团队主要是开发 NS 工程的团队, 因此, 需要调试 NS 工程, 需要能回退到 TZ-Closed 的权限, 当然完全回退权限也需要. 这些权限应该包含在其对应的一级证书内. 在生成 OEM 开发团队自己的证书之前, OEM 开发团队首先得拥有一套自己的公钥私钥对.

3.3.1. 生成自己的公钥私钥对



如上图所示, 用 TPC 打开 xml 文件 : C:/workspace/STM32Cube_FW_H5_V1.1.0/Projects/NUCLEO-H563ZI/ROT_Provisioning/DA/Config/DA_Config.xml

然后在 Debug Authentication root key 下, 点击 open, 打开 C:\workspace\STM32Cube_FW_H5_V1.1.0\Projects\NUCLEO-H563ZI\ROT_Provisioning\DA\Keys\key_2_intermediate.pem 这个一级密钥文件. 然后点击 regenerate 按键, 重新生成一个密钥对:



这个密钥就是 OEM 开发团队自己的密钥. 它的公钥可以发给安全开发团队, 用来生成一级证书和中间证书(请参考 3.2.4, 3.2.5 节).

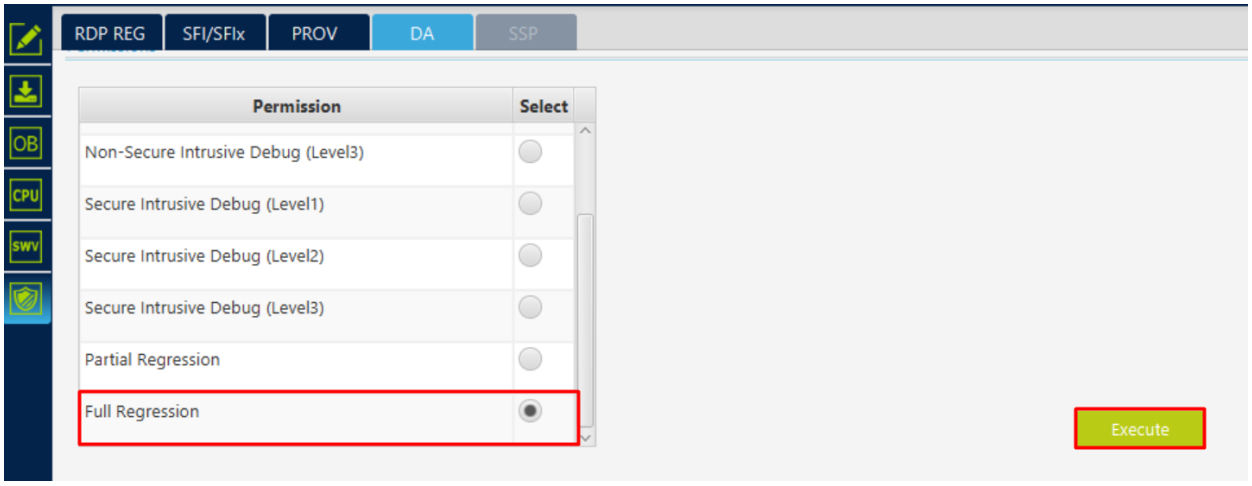
当然, 你也可以使用其它第三方工具, 比如 openssl 来生成自己的公钥私钥对, 最终以 pem 文件格式存在.

3.3.2. 测试一级证书的有效性

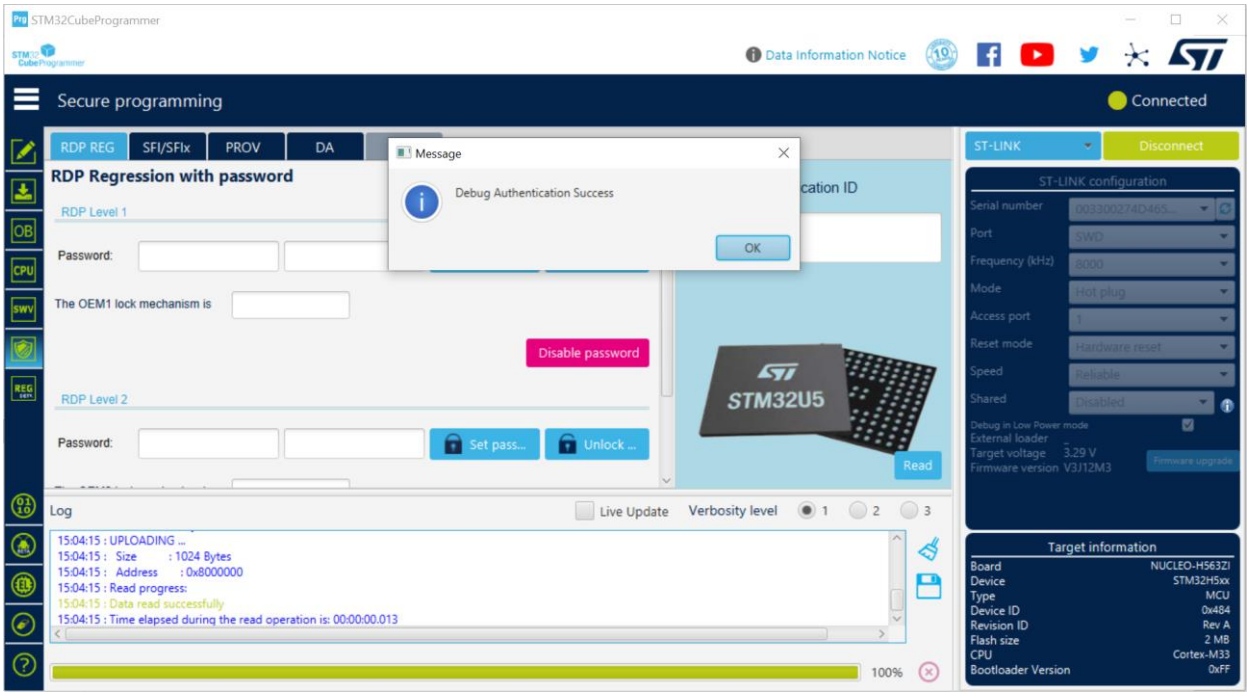
我们可以先在 provisioning 状态下测试下完全回退.

如上图所示, 输入 OEM 自己的私钥:
C:\workspace\STM32Cube_FW_H5_V1.1.0\Projects\NUCLEO-H563ZI\ROT_Provisioning\DA\Keys\key_2_intermediate.pem

同时输入一级证书: C:\workspace\STM32Cube_FW_H5_V1.1.0\Projects\NUCLEO-H563ZI\ROT_Provisioning\DA\Certificates\level1_cert_leaf_chain.b64



选择 Full Regression, 然后点击 Execute 按键...



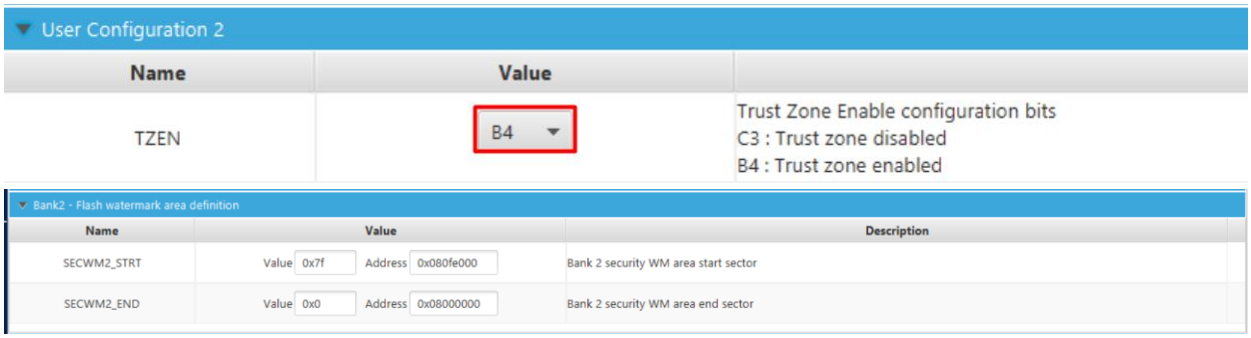
结果为可以完全回退成功.

接下来我们再测试下此证书是否可以调试 NS 工程...

在 Open 状态下, 我们烧一个测试程序:

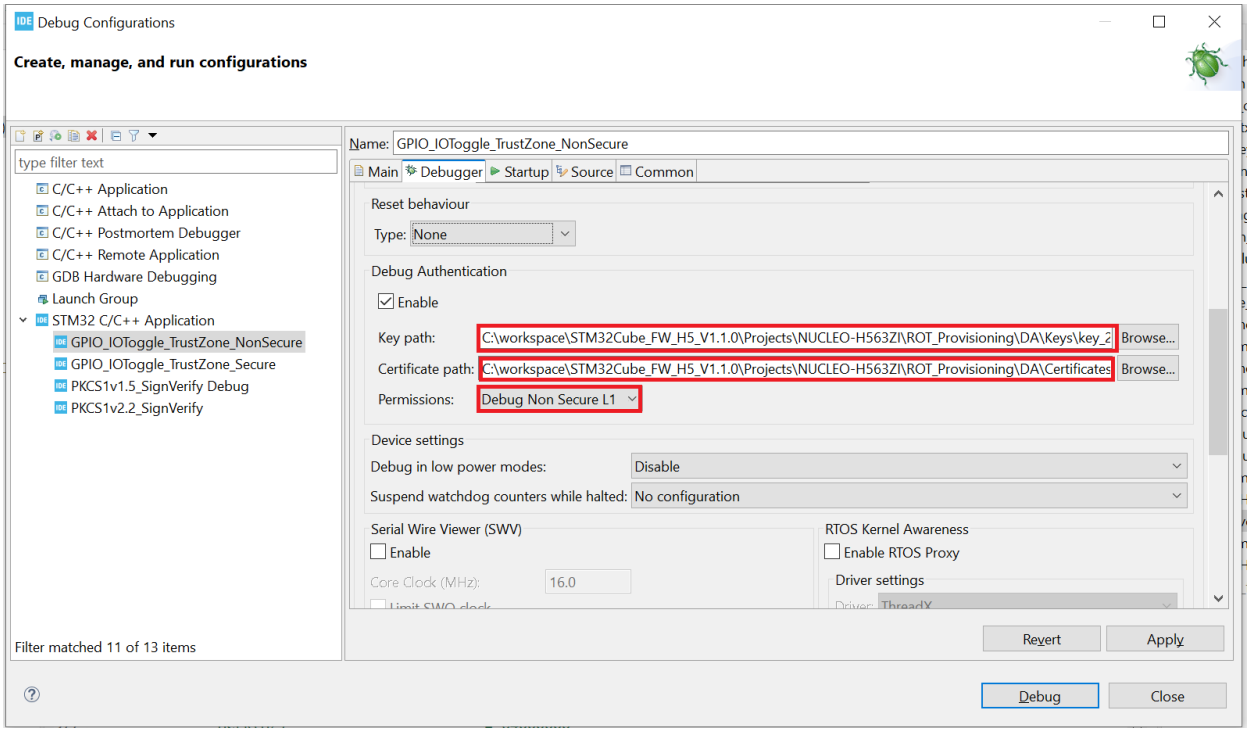
C:\workspace\STM32Cube_FW_H5_V1.1.0\Projects\NUCLEO-H563ZI\Examples\GPIO\GPIO_IOToggle_TrustZone

并修改芯片的选项字节 TZEN=0xB4, SECWM2_STRT=0x7f, SECWM2_END=0x0



烧录完固件后, 并确保此程序能正常运行(两个 LED 灯来回闪烁)的情况下, 再切换到 provisioning 状态下做 DA 预配置, 完了之后再切换到 closed 状态下. 然后我们再尝试使用 STM32CubeIDE 调试 NS 工程.

修改 NS 工程的调试配置...



如上图所示, 在 Key path 处输入 OEM 开发团队的私钥:

C:\workspace\STM32Cube_FW_H5_V1.1.0\Projects\NUCLEO-H563ZI\ROT_Provisioning\DA\Keys\key_2_intermediate.pem

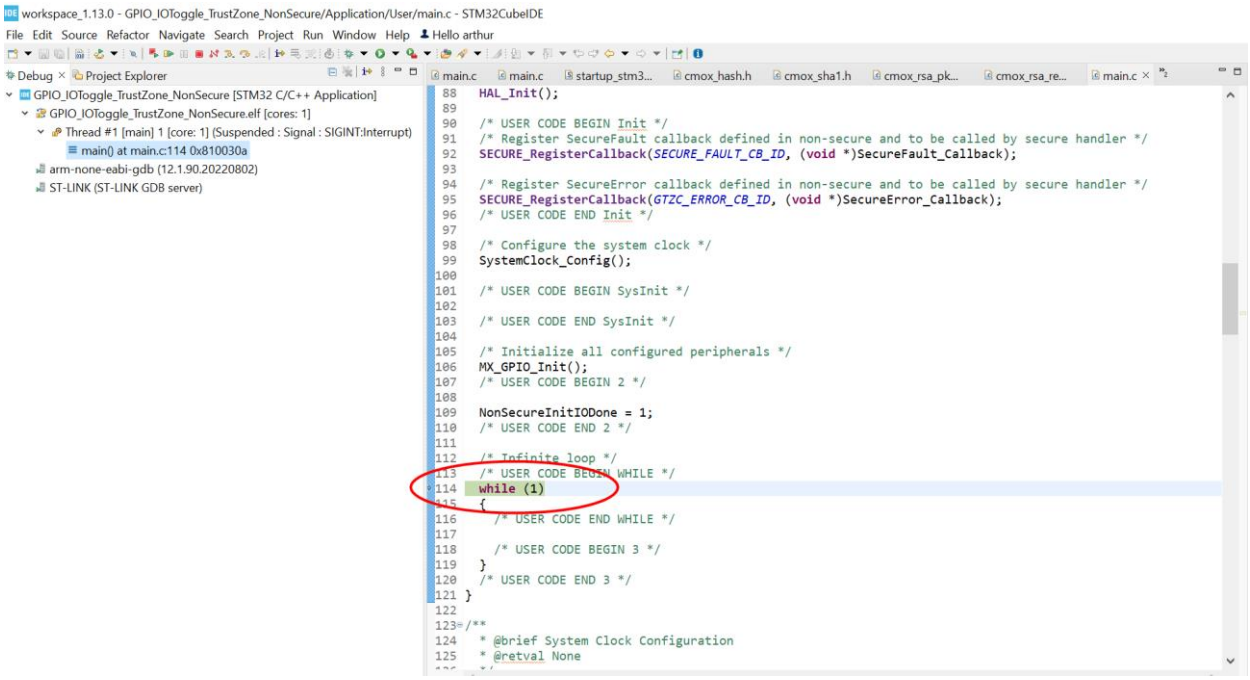
在 Certificate path 处输入一级证书:

C:\workspace\STM32Cube_FW_H5_V1.1.0\Projects\NUCLEO-H563ZI\ROT_Provisioning\DA\Certificates\level1_cert_leaf_chain.b64

Permissions 处选择 Debug Non Secure L1

Reset behavior 处选择 None.

然后点击 Debug 按键.. 之后在调试工具栏中点击暂停 ..



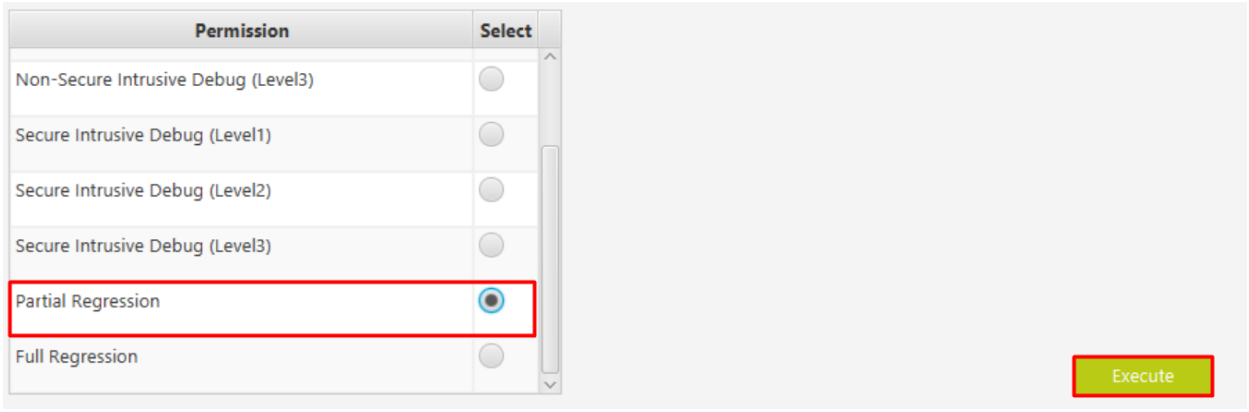
如上图所示, 程序在 NS 工程的 while(1);已经暂停了下来, 这说明使用一级证书调试 NS 工程是 OK 的.

接下来我们尝试使用一级证书进行局部回退.

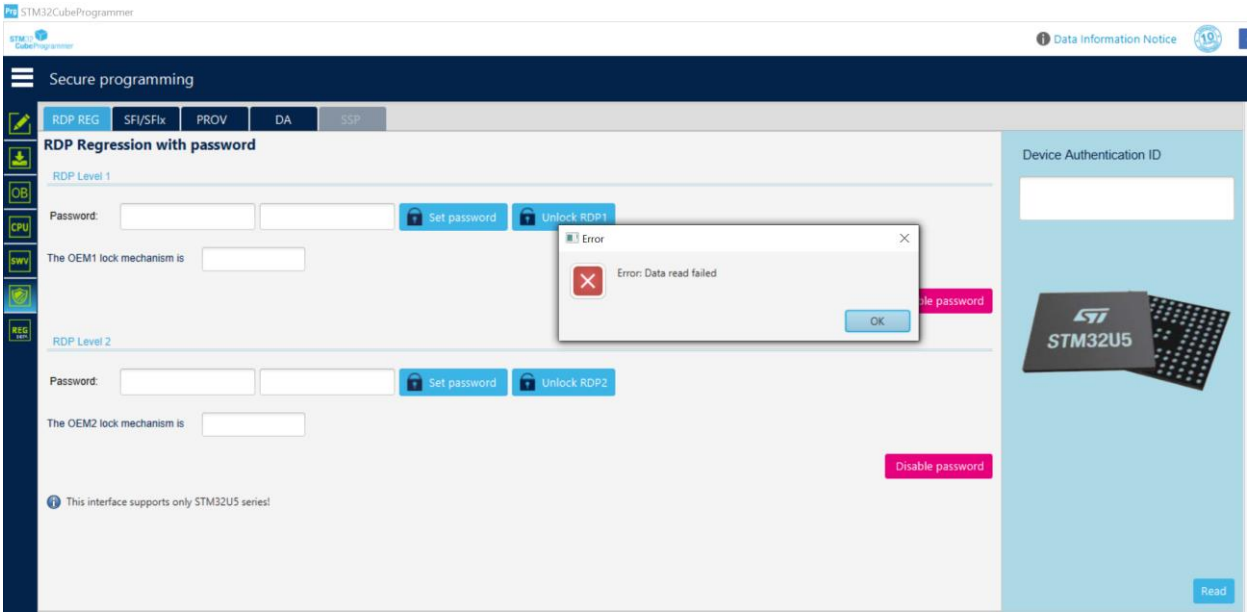
如上图所示,进 STLink 断开的情况下,点击 Discovery 后, 输入 OEM 开发团队的私钥:
C:\workspace\STM32Cube_FW_H5_V1.1.0\Projects\NUCLEO-H563ZI\ROT_Provisioning\DA\Keys\key_2_intermediate.pem

在下面输入框内输入一级证书:
C:\workspace\STM32Cube_FW_H5_V1.1.0\Projects\NUCLEO-H563ZI\ROT_Provisioning\DA\Certificates\level1_cert_leaf_chain.b64

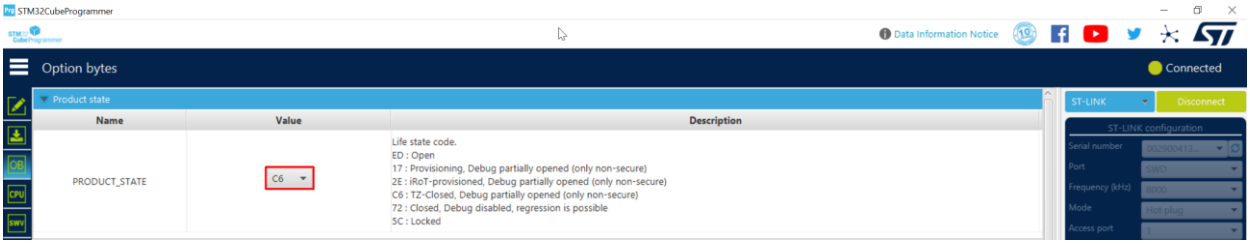
然后点击 Continue 按键...



如上图所示, 选择 Partial Regression, 然后点击 Execute 按键...



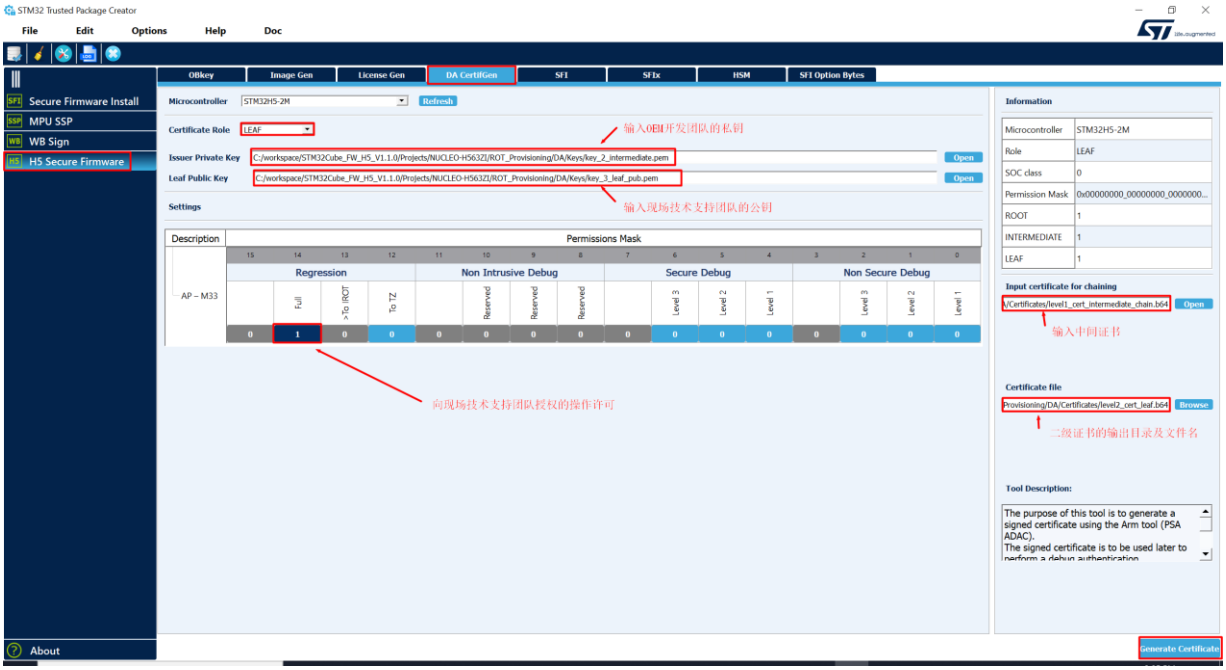
如上图所示, 执行成功, 并且 STLink 仍然保持连接, 此时查看状态为 TZ-Closed 状态:



这表明使用此一级证书进行半回退也是 OK 的.

3.3.3. 给现场技术支持团队生成二级证书

在给现场技术支持团队生成二级证书之前, 需要先拿到其公钥(见 3.4.1 节). 然后使 TPC 生成二级证书:



如上图所示, 在 Certificate Role 处选择 LEAF, 在 Issuer Private Key 处输入 OEM 开发团队的私钥: C:/workspace/STM32Cube_FW_H5_V1.1.0/Projects/NUCLEO-H563ZI/ROT_Provisioning/DA/Keys/key_2_intermediate.pem

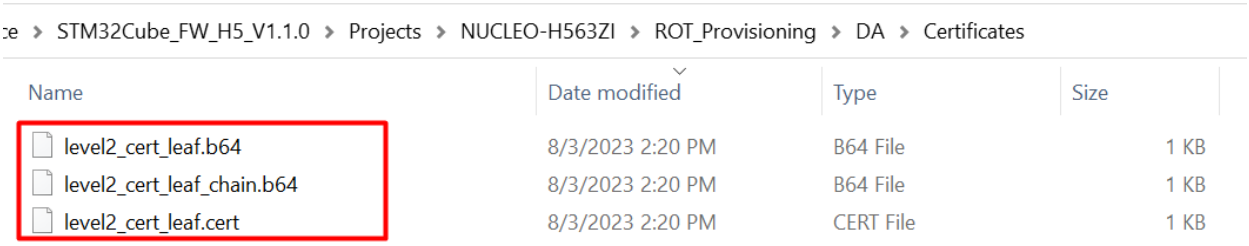
在 Leaf Poblic Key 处输入现场技术支持团队的公钥: C:/workspace/STM32Cube_FW_H5_V1.1.0/Projects/NUCLEO-H563ZI/ROT_Provisioning/DA/Keys/key_3_leaf_pub.pem

然后在 Settings 处选择给现场技术支持团队开放的权限. 需要注意地是, 由于中间证书都只有 Full Regression 的权限, 这里也就只能授权这个权限, 其它权限即便在这里点开了, 也不会有实际效果.

然后在右侧 Input certificate for chaining 处输入中间证书: C:/workspace/STM32Cube_FW_H5_V1.1.0/Projects/NUCLEO-H563ZI/ROT_Provisioning/DA/Certificates/level1_cert_intermediate_chain.b64

接着在 Certificate file 处输入需要导出的证书目录以及文件名: C:/workspace/STM32Cube_FW_H5_V1.1.0/Projects/NUCLEO-H563ZI/ROT_Provisioning/DA/Certificates/level2_cert_leaf.b64

最后点击 Generate Certificate 按键生成二级证书:



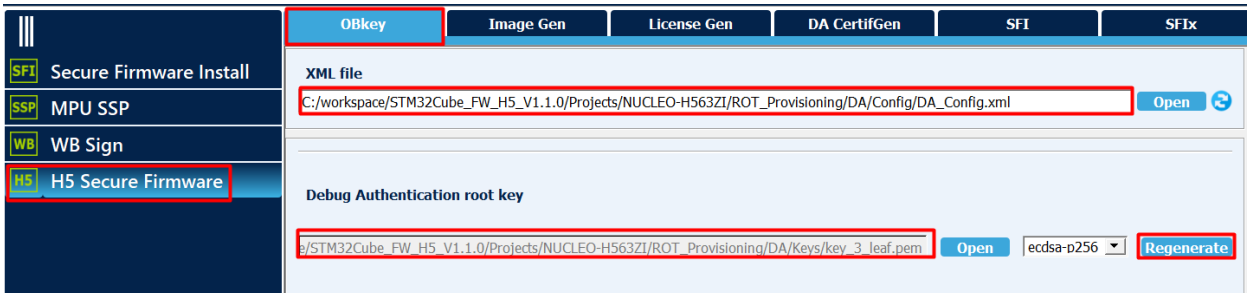
如上图所示, 生成的 level2_cert_leaf_chain.b64 就是二级证书, 就是需要发给现场技术支持团队用的.

3.4. 现场技术支持团队

现场技术支持团队是负责现场给客户进行技术支持的, 往往需要将芯片完全回退, 以便检查硬件是否出现问题. 因此, 需要完全回退的权限. 其证书仅仅需要此权限即可. 在生成对应的二级证书之前, 需要先拥有一对自己的公钥私钥对.

3.4.1. 生成自己的公钥私钥对

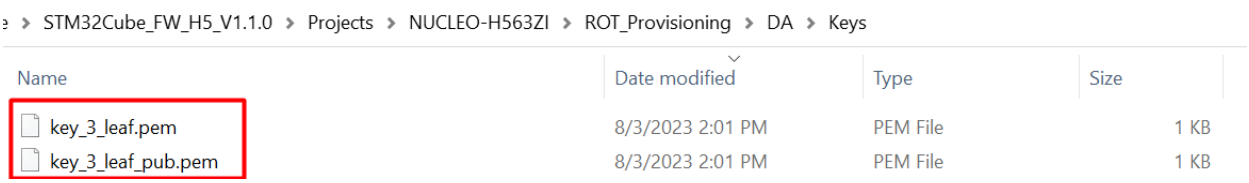
与 3.3.1 节类似, 使用 TPC 自己的公钥私钥对.



如上图所示, 在 XML file 处输入 C:/workspace/STM32Cube_FW_H5_V1.1.0/Projects/NUCLEO-H563ZI/ROT_Provisioning/DA/Config/DA_Config.xml

在 Debug Authentication root key 处, 点击 open 选择 C:\workspace\STM32Cube_FW_H5_V1.1.0\Projects\NUCLEO-H563ZI\ROT_Provisioning\DA\Keys\key_3_leaf.pem

最后点击 Regenerate 按键生成密钥对:



如上图, 其中的 key_3_leaf_pub.pem 文件为公钥, 需要发给 OEM 开发团队用来生成二级证书(见 3.3.3 节).

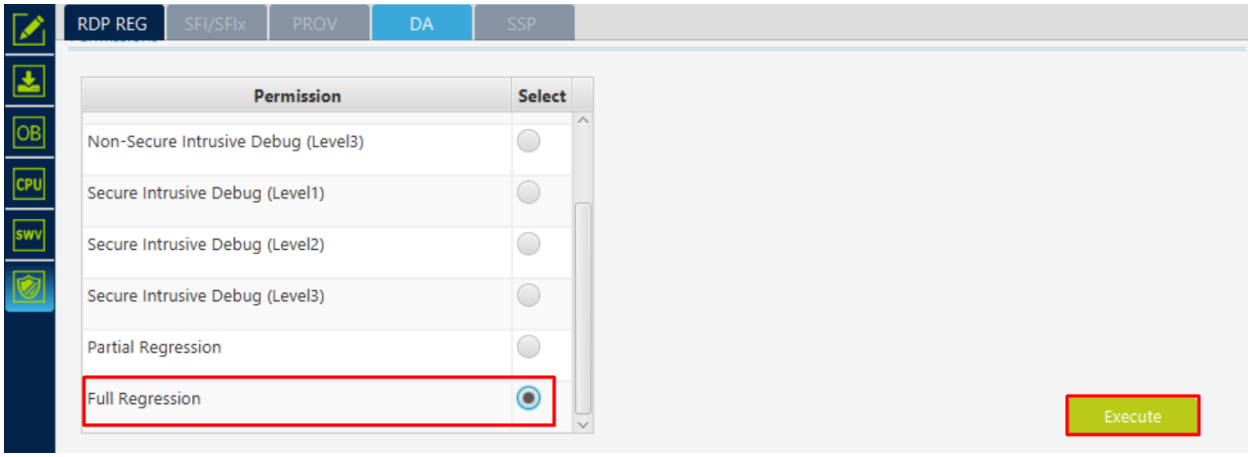
3.4.2. 测试二级证书的有效性

之前我们已经将 NUCLEO-板回退到 TZ-Closed 状态, 可直接在此状态下使用二级证书进行完全回退, 以验证二级证书的有效性.

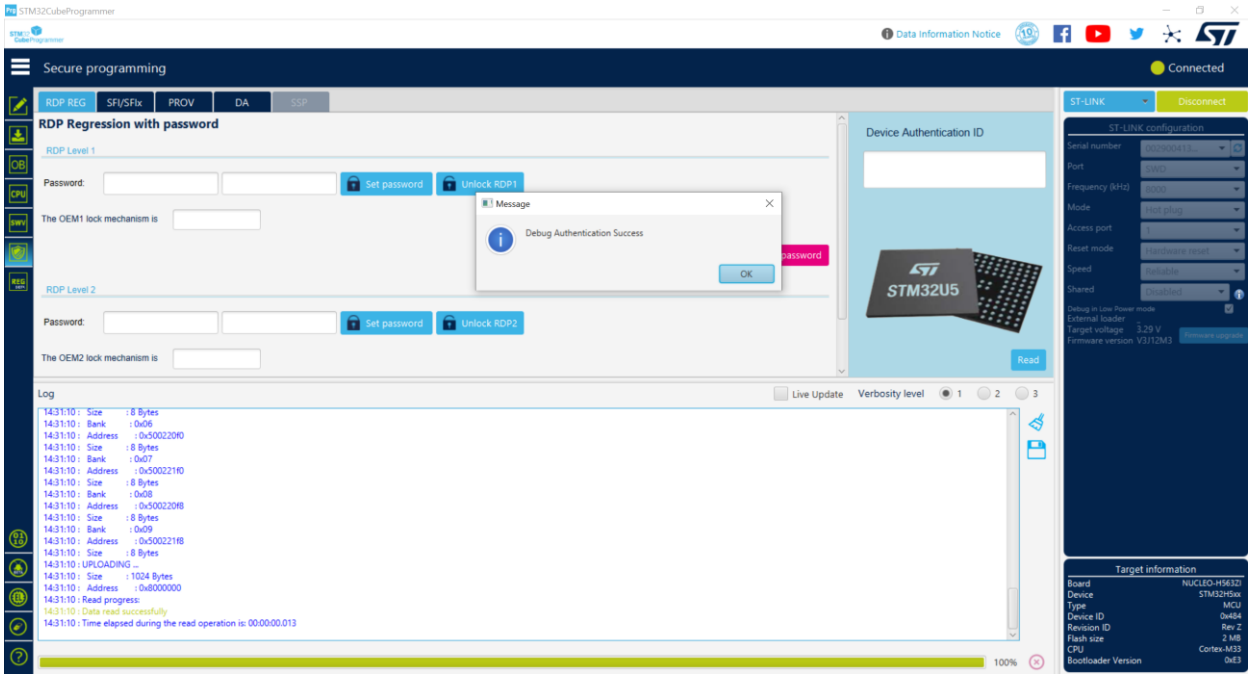
如上图所示, 在 STM32CubeProgrammer 中, 在 ST-Link 断开的情况下, 点击 Discovery 按键, 然后在 key File path 处输入现场技术支持团队自己的私钥:
C:\workspace\STM32Cube_FW_H5_V1.1.0\Projects\NUCLEO-H563ZI\ROT_Provisioning\DA\Keys\key_3_leaf.pem

在 Certificate File Path 处输入二级证书:
C:\workspace\STM32Cube_FW_H5_V1.1.0\Projects\NUCLEO-H563ZI\ROT_Provisioning\DA\Certificates\level2_cert_leaf_chain.b64

然后点击 Continue 按键...



然后选择 Full Regression, 最后点击 Execute 按键...



最终完全回退成功.

这就验证了此二级证书的有效性.

至此, 根证书, 一级证书, 二级证书均已验证其有效性.

4. 其它问题

在生成证书链过程中, 有涉及到中间证书, 我们用它来生成二级证书, 原则上中间证书只能用来生成二级证书, 其本身并不会直接使用, 那么这里有一个问题, 直接使用中间证书能有效吗? 我们不妨来测试下.

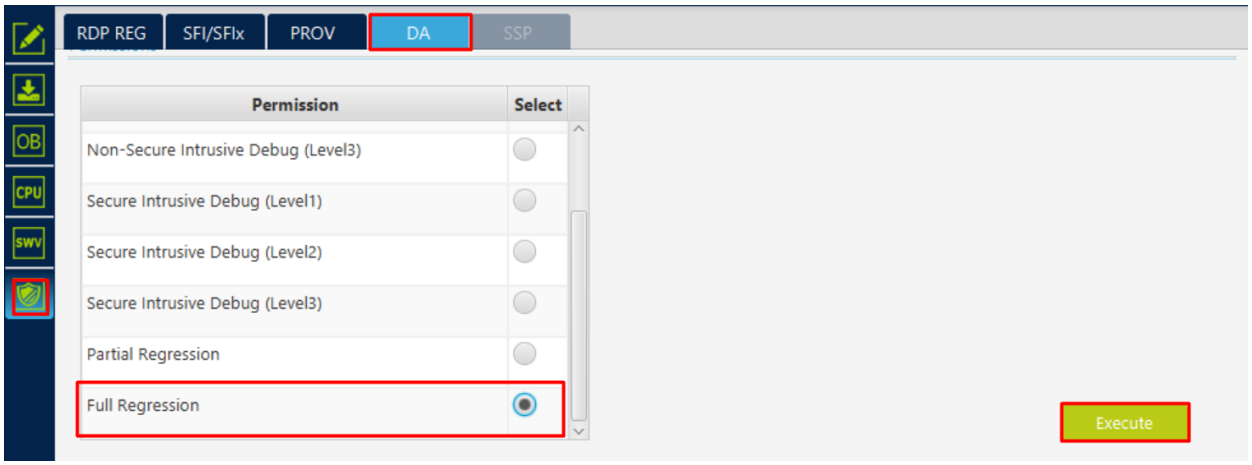
4.1. 测试直接使用中间证书

在芯片 provisioning 状态下, 我们给芯片做好预配置后, 可直接使用中间证书尝试完全回退.

如上图, 在 Key File Path 处使用 OEM 开发团队的私钥: C:\workspace\STM32Cube_FW_H5_V1.1.0\Projects\NUCLEO-H563ZI\ROT_Provisioning\DA\Keys\key_2_intermediate.pem

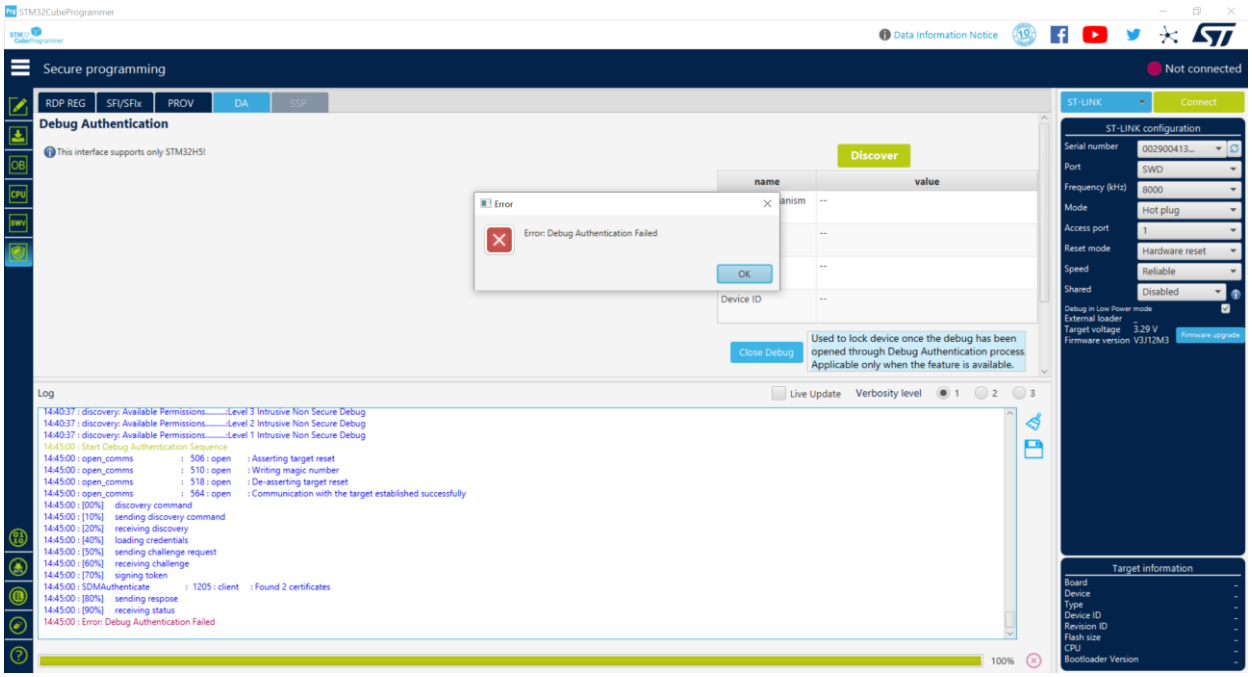
然后证书处输入中间证书: C:\workspace\STM32Cube_FW_H5_V1.1.0\Projects\NUCLEO-H563ZI\ROT_Provisioning\DA\Certificates\level1_cert_intermediate_chain.b64

然后点击 Continue 按键...



然后选择 Full Regression, 最后点击 Execute 按键...

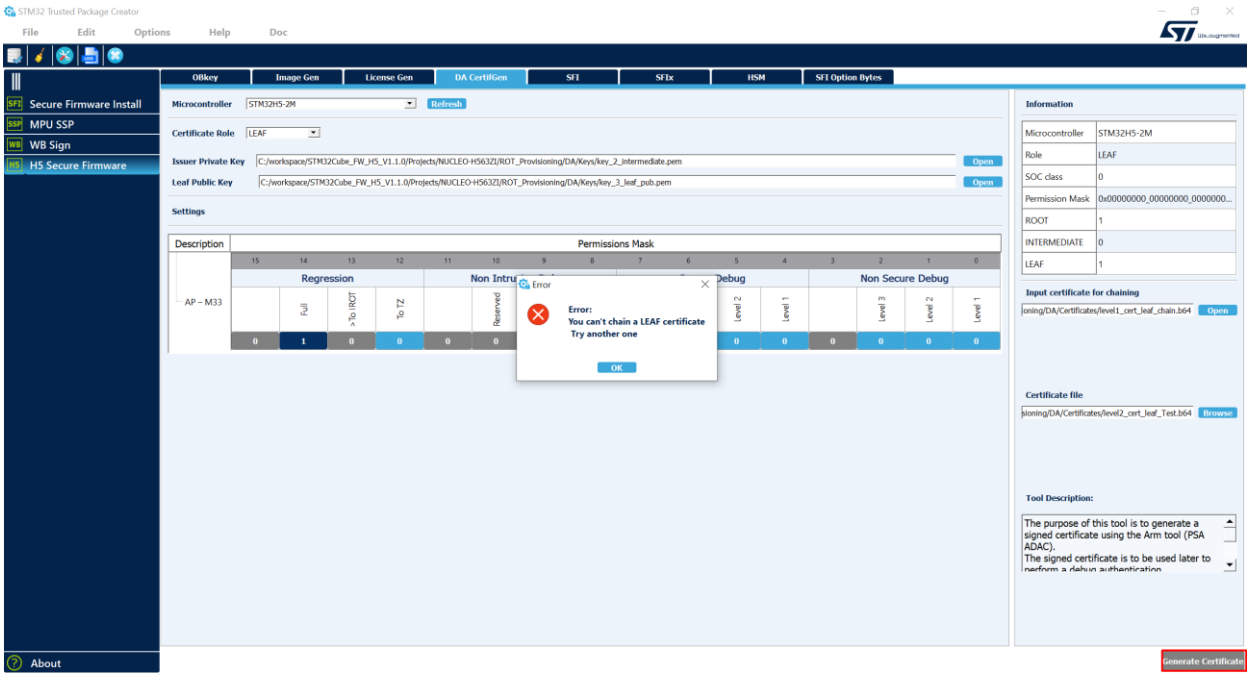
之前在生成中间证书时, 是给中间证书授权完全 回退的, 接下来看看实际测试结果...



如上图所示, 实际测试结果为失败. 由此可见, 中间证书确实是不能直接拿来使用的.

4.2. 测试使用一级证书生成二级证书的有效性

之前我们是使用了中间证书来生成二级证书, 如果我们直接使用一级证书生成二级证书, 会怎样? 这个二级证书是否仍然有效? 我们接着来测试下.



如上图所示, 在使用 TPC 生成二级证书过程中, 在右边的 Certificate file 处输入一级证书 C:/workspace/STM32Cube_FW_H5_V1.1.0/Projects/NUCLEO-H563ZI/ROT_Provisioning/DA/Certificates/level1_cert_leaf_chain.b64

时, 会直接弹出错误提示框, 且右下角 Generate Certificate 按键也是灰色的, TPC 不允许这么操作. 实测通过 TPC 是无法生成这种证书的.

版本历史

日期	版本	变更
2023 年 10 月 17 日	1.0	首版发布

重要通知 - 请仔细阅读

意法半导体公司及其子公司 （“ST”）保留随时对 ST 产品和 / 或本文档进行变更的权利，恕不另行通知。买方在订货之前应获取关于 ST 产品的最新信息。 ST 产品的销售依照订单确认时的相关 ST 销售条款。

买方自行负责对 ST 产品的选择和使用， ST 概不承担与应用协助或买方产品设计相关的任何责任。

ST 不对任何知识产权进行任何明示或默示的授权或许可。

转售的 ST 产品如有不同于此处提供的信息的规定，将导致 ST 针对该产品授予的任何保证失效。

ST 和 ST 徽标是 ST 的商标。若需 ST 商标的更多信息，请参考 *www.st.com/trademarks*。所有其他产品或服务名称均为其 各自所有者的财产。

本文档是 ST 中国本地团队的技术性文章，旨在交流与分享，并期望借此给予客户产品应用上足够的帮助或提醒。若文中内容存有局限或与 ST 官网资料不一致，请以实际应用验证结果和 ST 官网最新发布的内容为准。您拥有完全自主权是否采纳本文档（包括代码，电路图等）信息，我们也不承担因使用或采纳本文档内容而导致的任何风险。

本文档中的信息取代本文档所有早期版本中提供的信息。