RTL8762D Security Mechanism User Guide

V0.1

2020/06/04

Revision History

Date	Version	Modification	Author	Reviewer
2020/06/04	Draft v0.1	First edition	Serval Li	

Content

Table List

Figure List

1 Introduction

This paper introduces security mechanism of RTL8762D as well as its usage. Security mechanism protects images in Flash by encrypting data, and it also includes downloading decryption key and controlling debug port.

2 Security Mechanism

Security mechanism includes image encryption, eFuse key, SWD interface control and Password debug.

2.1 Image Encryption

Encryption is mandatory to Patch image and optional to APP image. AES symmetric encryption algorithm is used to encrypt the images and the encryption key has the length of 128 bits. When IC is booting, image will be decrypted by reading the key in eFuse. If key is not downloaded or the key downloaded doesn’t match the encryption key, the boot process will fail.

2.2 eFuse Key

It is necessary to find out a special mechanism to protect encryption key from leakage for the reason that Encryption and decryption use the same 128-bit key. A new key will be generated when passing Encryption key to Encryption Tool, which will also be published and downloaded into eFuse of IC. During the download process, Download Tool will decrypt the new key to obtain original key and read UUID of IC at the same time. These information will be combined together to generate a new key to ensure that each IC has a unique key.

2.3 SWD Interface Control

SWD interface is an important debug port that plays a vital role in debugging program. However, it also increases the risk of exposing data and code. Security mechanism provides 3 methods to control SWD interface: Open, Close and Password Control, among which Password Control denotes the SWD interfaces is always closed unless correct password is received through UCI UART.

2.4 Password Debug

Similar to encryption key, password is also programmed in eFuse of IC. When password is received through HCI UART, IC will reboot automatically to check if the password is correct. The function configured as Password control is always closed unless the password is correct. Each time IC reboots, password need be retyped to active the function.

3 Security Level

RTL8762D provides 3 security levels: 0, 1 and 2. Larger number indicates higher security level, which will affect debug and re-program of eFuse. Function control of each module under different security level is listed in Table 3-1. It is suggested to configure security level to level 1 during minor trial-production and level 2 in mass production.

Table 3-1 Security Level Configuration

Security Level	SWD Control	eFuse Read	eFuse Write	HCI Download	HCI BT Test
0	Enable	Enable	Enable	Enable	Enable
1	Enable by password	Enable by password	Enable	Enable	Enable
2	Enable by password	Disable	Enable by password	Enable	Enable

4 Usage Example

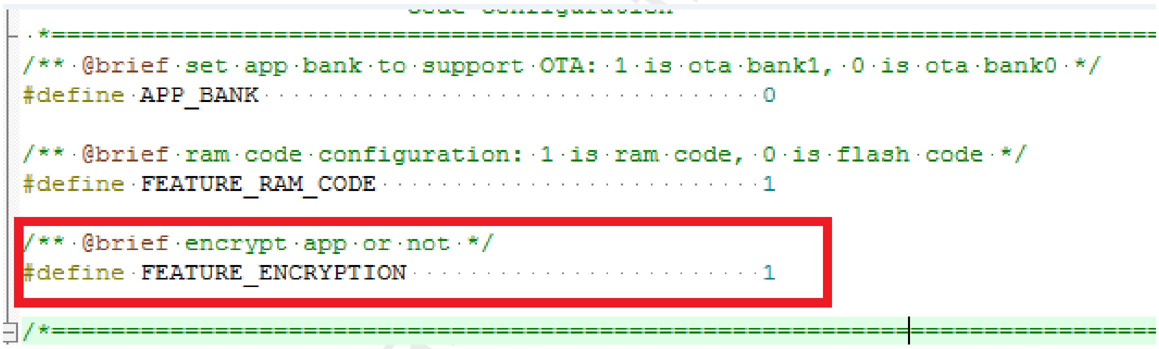
4.1 Configure Encryption Key

Edit JSON file located at sdk\tool\key.json to configure OCEK and PASSWORD, where OCEK and PASSWORD are plaintext that needs protection.

```
{
    "OCEK":    "a1a2a3a4a5a6a7a8a9aaabacadaeafb0",
    "PASSWORD":  "00112233445566778899aabbccddeeff"
}
```

4.2 Generate Encrypted APP Image

Attach APP_ENCRYPTION_TEXT_SECTION to the function to be encrypted. In mem_config.h of SDK, Macro FEATURE_ENCRYPTION determines if the APP requires encryption. It is assigned to 0 by default, which indicates not encrypted.



```
 *=============================================================================
/** @brief set app bank to support OTA: 1 is ota bank1, 0 is ota bank0 */
#define APP_BANK                                 0

/** @brief ram code configuration: 1 is ram code, 0 is flash code */
#define FEATURE_RAM_CODE                         1

/** @brief encrypt app or not */
#define FEATURE_ENCRYPTION                       1

/*==============================================================================
```

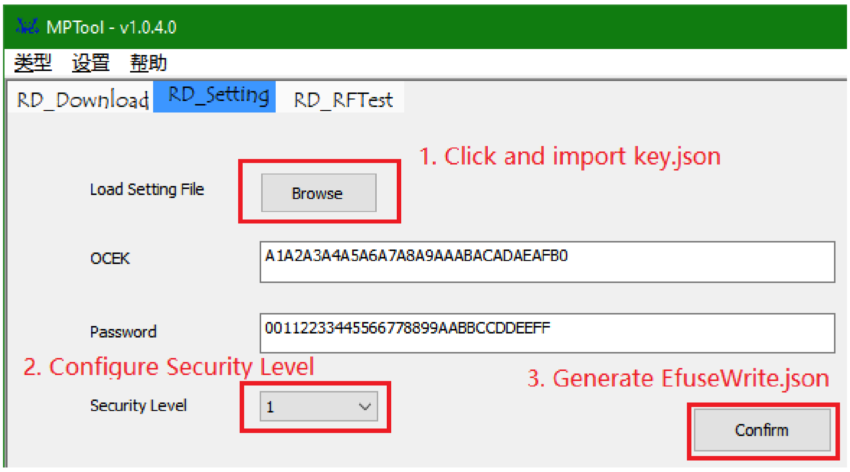
Figure 4-1 Encrypt APP Code

4.3 Program eFuse

Note:

2.5V (±10%) power supply must be applied when programming eFuse. The download procedure introduced in this document applies to Flash that supports wide voltage range and can be powered by 2.5V (±10%) power supply. Under such circumstance, Flash and eFuse can be programmed in one step.

RD end provides the eFuse file to be programmed.


MPTool - v1.0.4.0
类型 设置 帮助
RD_Download
RD_Setting
RD_RFTest
Browse
Confirm
1. Click and import key.json
2. Configure Security Level
3. Generate EfuseWrite.json

Load Setting File

OCEK: A1A2A3A4A5A6A7A8A9AAABACADAEAFB0

Password: 00112233445566778899AABBCCDDEEFF

Security Level: 1

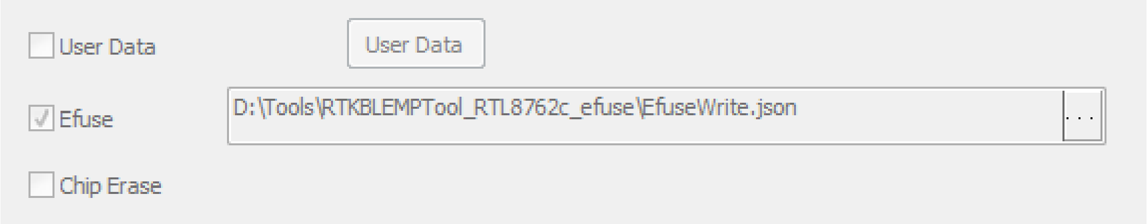
Figure 4-2 Generate the File to Program eFuse

Above all, confirm that MP Tool is in debug mode: Click ‘Type’ button on tool bar and tick ‘Debug’.

1. Click “Browse” button to import key.json file in ”RD Setting” UI;
2. Select appropriate Security Level for project;
3. Click “Confirm” button” to generate EfuseWrite.json file, which can be released to factory for programming eFuse.

Program eFuse in factory

Above all, confirm that MP Tool is in mass production mode: Click ‘Type’ button on tool bar and tick ‘MP



☐ User Data

☑ Efuse: D:\Tools\RTKBLEMPTool_RTL8762c_efuse\EfuseWrite.json

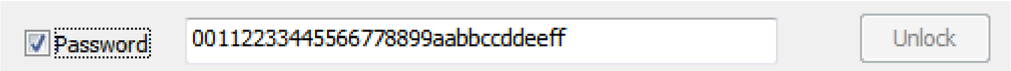
☐ Chip Erase

Figure 4-3 Select the File to be Prgrammed in eFuse

1. Tick “Efuse” in “MP Setting” UI and select the eFuse file to be downloaded;
2. Click “Download” button in “MP Download” UI to program eFuse.

4.4 Password Debug

When security level is 1 or 2, SWD interface will be banned and developer can reactive SWD interface by Password debug.



☑ Password: 00112233445566778899aabbccddeeff

Figure 4-4 Use PASSWORD to unlock SWD

Open serial port in “RD Download” interface and tick “Password”. Type in the plaintext of password defined in key.json and then click “Unlock” button, IC will reboot. SWD interface will be reactivated after reboot process.