The secure\_files example application demonstrates reading and writing encrypted files from/to the filesystem. To use secure\_files, the secureboot mode should be enabled.

For more details on secureboot mode, refer Application\_for\_using\_SSBL.pdf

# Description of Operation

1. Reconstruct secureboot secret and get the firmware cipher key from flash
2. Mount the filesystem
3. Write the encrypted message to a file using the firmware cipher key
4. Read the encrypted message as plain text (should be garbled)
5. Read and decrypt encrypted message (this should show the original message contents)

# Building and Flashing Process

## Building Components

1. Build secureboot SSBL for emulation usecase (both.img).

(refer section: 6.2.2 in Application for\_using\_SSBL.pdf*)*

1. Build the secure\_files example:

|  |
| --- |
| cd <freertos\_sdk>/examples/secure\_files  make clean  make KEY=../../apps/ssbl/enroll.json |

This creates a signed and encrypted application binary examples/secure\_files/out/secure\_files.elf.enc

1. Build the secure filesystem (root\_secure.img):

(refer section: 6.1.2 in Application for\_using\_SSBL.pdf*)*

|  |
| --- |
| cd <freertos\_sdk>  python ./script/build\_rootfs\_generic.py --folder\_path examples/secure\_files/ --secure True --keyfile ./apps/ssbl/enroll.json |

This creates a root image binary examples/secure\_files/root\_secure.img

## Flashing Components

1. Enroll keys & flash SSBL components in secureboot mode for emulation usecase.

(refer steps 1 to 5 of section: 7.2.1 in Application for\_using\_SSBL.pdf*)*

1. Flash application at 0x20000

|  |
| --- |
| cd <freertos\_sdk>  ./script/flash.py --device /dev/ttyUSB2 write 0x20000 ./examples/secure\_files/out/secure\_files.elf.enc |

1. Flash the filesystem at 0x180000

|  |
| --- |
| cd <freertos\_sdk>  ./script/flash.py --device /dev/ttyUSB2 write 0x180000 ./examples/secure\_files/root\_secure.img |

1. Reset the board

Reset the board either by executing the following command or by pressing the reset button on the EVB to run secure\_files application.

|  |
| --- |
| cd <freertos\_sdk>  ./script/boot.py --device /dev/ttyUSB2 --reset=evk42 |

## Expected Output

When DEBUGSECURE=1

|  |
| --- |
| Y-BOOT 208ef13 2019-07-22 12:26:54 -0500 790da1-b-7  ROM yoda-h0-rom-16-0-gd5a8e586  FLASH:PNWWAE  FIRST:SWWWWAHE  Si  Build $Id: git-a74c874 $  Flash detected. flash.hw.uuid: 39483937-3207-0051-002a-ffffffffffff  \*\*\*Warning! Make sure to remove this code section once in production\*\*\*  secureboot\_secret:  8b5678a045ba66b7ea956d3292aae8dc29ded8de9010efd40980a091734b786b11000000  \*\*\*Warning! Make sure to remove this code section once in production\*\*\*  cipher key: 4e3b0b9792183c53ecc78a38c64a45c071b97bc40b0baba308ed76db8a46cef1  public key: 20b003d2f297be2c5e2c83a7e9f9a5b9eff49111acf4fddbcc0301480e359de6dc809c49652aeb6d63329abf5a52155c766345c28fed3024741c8ed01589d28b  Build $Id: git- a74c874 $  Flash detected. flash.hw.uuid: 39483937-3207-0051-002a-ffffffffffff  Bootargs: vm.flash\_location=0x0002d900 passphrase=12346789ssid=innotest  sys.reset\_reason=1  Application Information:  ------------------------  Name       : Secure files demo application  Version    : 1.0  Build Date : Aug 26 2023  Build Time : 18:50:21  Heap Available: 402 KB (411896 Bytes)  Original message: Hello! This is a plain text file.  Writing message to encrypted file  Reading file as ciphertext  Cipher text message: 1~␒M}rQo앺{AÛ␒\*\_/rY0  Reading and decrypting file  Plain text message: Hello! This is a plain text file. |