Hello, and welcome to this presentation describing the Secure Firmware Install (SFI) feature offered by the Root Security Services.

Secure Firmware Install (SFI) for STM32H7RS

SFI Definition

- Secure mechanism implemented in STM32 secure bootloader to allow secure and counted installation of OEM firmware
- SFI process prevents the OEM firmware code from:
  - Being accessed by the contract manufacturer
  - Being extracted or disclosed
- SFI consists in having the following content encrypted with an AES secret key and safely installed :
  - The OEM firmware to embed into the STM32H7RS
  - The Option Bytes Keys (OBK)
  - The Option Bytes

RSS/RSSe

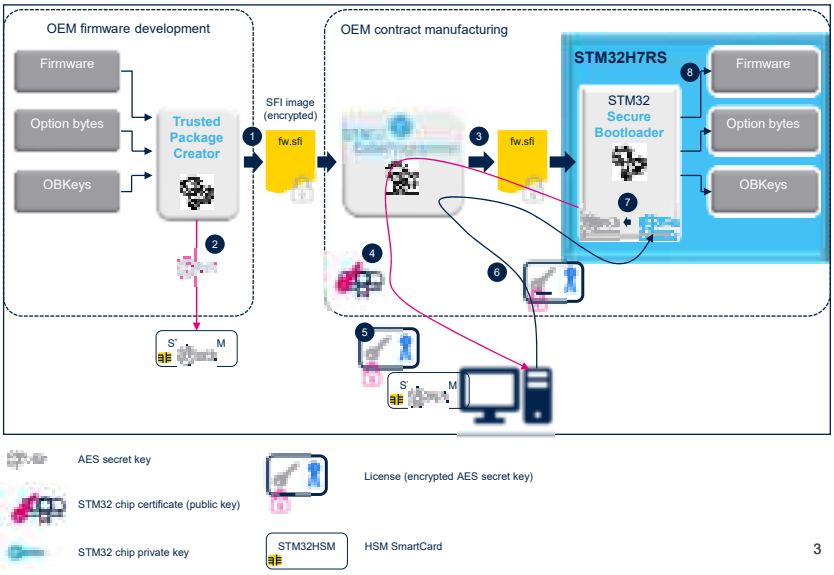
- RSS stands for Root Security Services and RSSe for RSS extension
- Provide essential security services for STM32 to ensure the integrity, confidentiality, and authenticity of firmware and data:
  - Secure Boot: Ensures that only authenticated firmware can be executed
  - Cryptographic Operations: Handles encryption, decryption, and key management
  - Secure Storage: Protects critical data such as secret keys from unauthorized access
  - Debug Authentication: Controls access to the debug interface based on authentication mechanisms

2

Outsourcing of product manufacturing enables Original Equipment Manufacturers (OEMs) to reduce their direct costs and concentrate on high added-value activities, such as research and development, sales and marketing.
However, contract manufacturing puts the OEM's proprietary assets at risk, and since the Contract Manufacturer (CM) manipulates the OEM's Intellectual Property (IP), it might be disclosed to other customers, or appropriated.
To meet the new market security requests and protect customers against any leakage of their IPs, STMicroelectronics Secure Firmware Install (SFI) allows programming of OEM firmware into STM32 MCU internal flash memory in a secure way (with confidentiality, authentication and integrity checks).

STM32H7RS SFI Process

1. SFI image (encrypted) available from *STM32 Trusted Package Creator*
2. The OEM programs the HSM with the AES secret key
3. Start of the SFI process
4. Device certificate retrieval through the STM32 CubeProgrammer
5. The HSM generates the license based on the certificate
6. The HSM provides the license to the STM32 through the STM32 CubeProgrammer
7. The RSS/RSSe retrieve the OEM AES secret key encrypted in the license
8. The encrypted firmware and option bytes are decrypted then programmed



3

The secure bootloader is a standard ST bootloader with additional security features.
If the STM32 microcontroller is reset during retrieving AES secret key, all sensitive data is erased before restarting initial SFI procedure.
During SFI process, the secure bootloader never allows any other code to access user flash memory or SRAM.

The installation of secure firmware in internal Flash memory is achieved through the following steps:

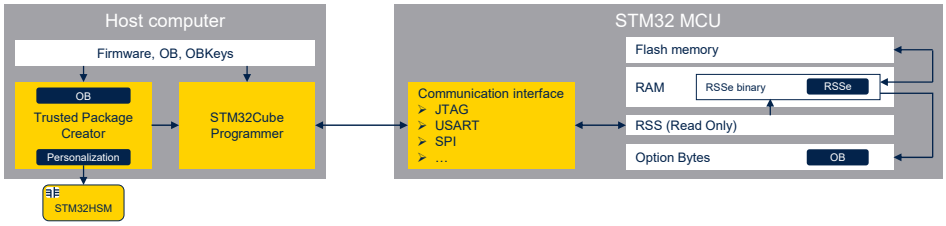
1- The (encrypted) SFI image is available from STM32 Trusted Package Creator
2- The OEM programs the Hardware security module (HSM) with the AES secret key
3- Start of the SFI process

4- Device certificate retrieval
5- STM32 device authentication in the HSM
6- The HSM provides the license to the STM32 device
7- The RSS retrieves the OEM AES secret key encrypted in the license
8- Encrypted firmware and option bytes are transferred, decrypted then programmed.

Note that three keys are used in this sequence:
The AES symmetric key
The STM32chip private and public asymmetric keys.

SFI toolsets

- ST SFI offers complete toolsets
  - STM32 Trusted Package Creator software package to encrypt OEM binaries
  - The STM32CubeProgrammer to securely flash the STM32
  - The STM32-HSM to transfer OEM credentials to the programming partner
- User can refer to X-Cube-RSSe, X-Cube-SFI and AN4992



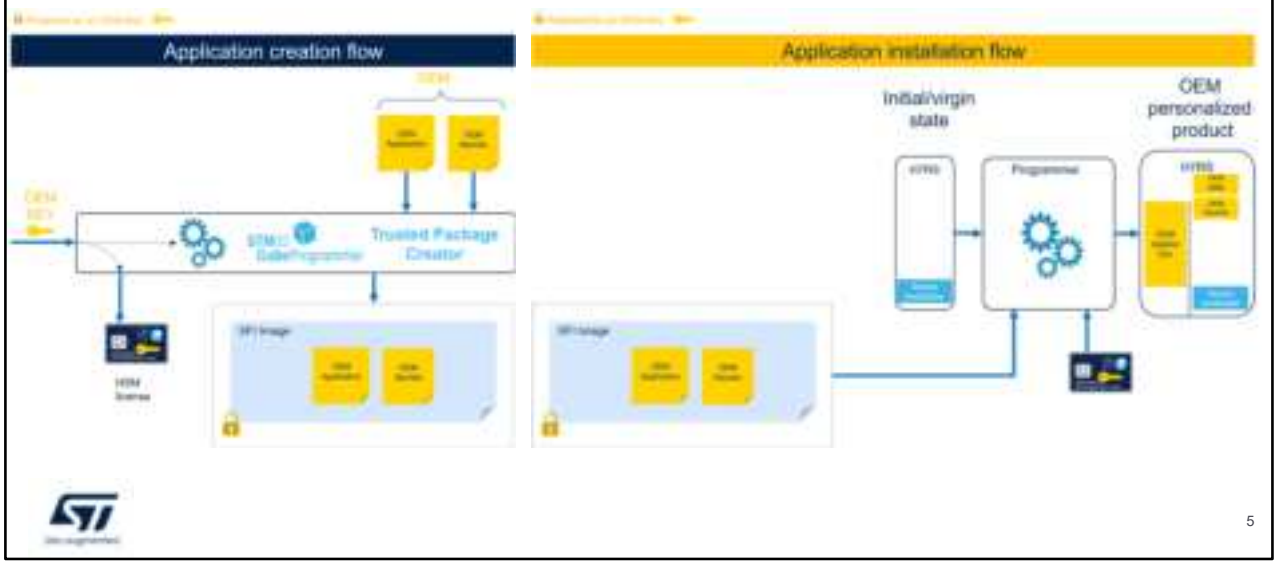
4

The STM32 secure bootloader, implementing SFI, is programmed during STM32 manufacturing. It manages communication between STM32CubeProgrammer and STM32 device to:
Identify STM32 device
Exchange device certificate and license
Download SFI encrypted image inside STM32.

To prepare the SFI image and provision the STM32, ST SFI offers complete toolsets :
STM32 Trusted Package Creator software package to encrypt OEM binaries,
The STM32CubeProgrammer to securely flash the STM32 device,
The STM32-HSM to transfer OEM credentials to the programming partner.

User can refer to AN4992, X-cube-RSSe and X-Cube-SFI readme files for a step-by-step SFI process.

SFI Flow



This slide explains the various steps of the application creation flow.

Once the OEM application has been developed, the OEM prepares and tests the SFI image to be installed during manufacturing. For that purpose, the OEM must use the STM32 Trusted Package Creator tool.

This allows the correct generation of the SFI image and its testing before manufacturing. The output of the STM32 Trusted Package Creator is the tested SFI image, ready to be installed during manufacturing.

The purpose of this step is to:

- Prepare the encrypted firmware image to install, called the SFI image. It is composed of the OEM application and the additional components (OEM secrets and OEM option bytes).
- Provision the OEM key within an HSM.

The application installation flow is like the generic SFI installation procedure, which is deployed on other STM32 products supporting SFI.

- The main difference is that the interaction between the host and the STM32H7RS MCU is not possible during SFI.
- SFI on external flash memory (SFIx) is not supported on STM32H7RS MCUs.

Application creation flow: SFI image inputs/Outputs

SFI image Inputs	SFI image Output

• The OEM application: The OEM must provide its application binary
• OEM secrets are the OEM data and the OEM keys
  • OEM option byte key (OBKey) provisioning:
  • OBKey HDPL0 must be done first; It includes debug authentication (DA) configuration
  • Then, other OBKey can be done (optional), such as OBKey iRoT
• OEM option bytes (OB): The OEM must set carefully the STM32H7RS product state
• The STM32H7RS flash configuration to install via the SFI procedure must be the same than the one used during the OEM application development
	• The Trusted Package Creator encrypts the SFI image inputs with the OEM key and generates the SFI image
	• The SFI image is then an encrypted image containing the OEM application, the OEM secrets, and the OEM option bytes

6

SFI image inputs according to the SFI flow in previous slide are as follows:
The OEM application
OEM secrets
OEM Option Bytes (OB)
The STM32H7RS flash configuration to install via the SFI procedure, that must be the same than the one used during the OEM application development.
The SFI image output according to the SFI flow in previous slide is the SFI image, which is an encrypted image.

OEM key provisioning

- The OEM must provide its OEM key to the Contract Manufacturer (CM) in such a way that the OEM key cannot be read or extracted clearly by the CM
  - Only the STM32 can handle the OEM key
- In SFI for STM32H7RS, the OEM provisions its OEM key, using the Trusted Package Creator, in one HSM
  - This must be done before providing the OEM key to the CM
- Advantages of this provisioning process:
  - Only the STMicroelectronics STM32 microcontrollers can securely install the SFI image
  - The authenticity, integrity, and confidentiality of the SFI image content are ensured
  - When using the HSM, the number of STM32 chips to program can be counted

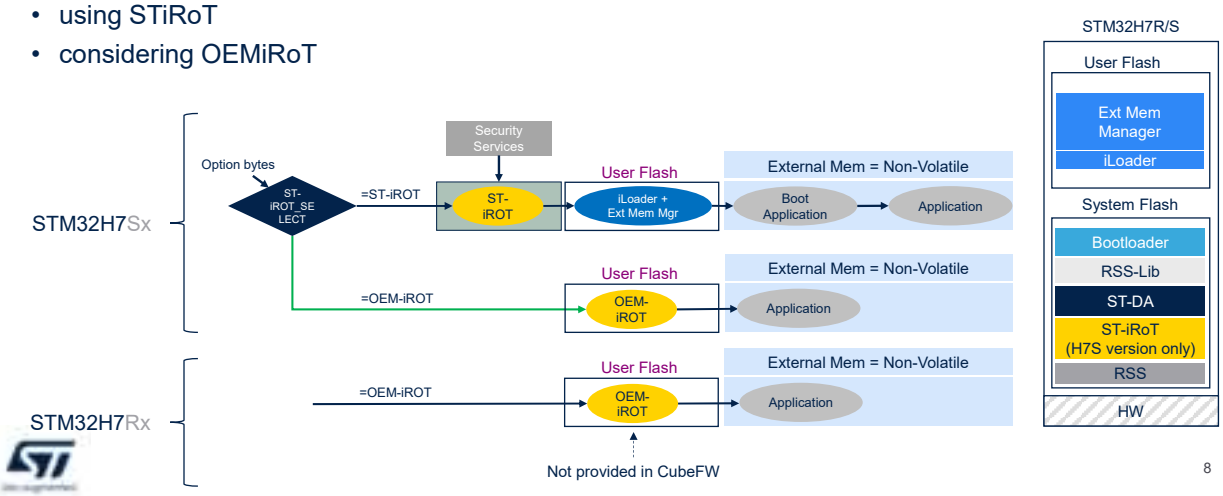
7

The OEM must provide its OEM key to the Contract Manufacturer (CM) in such a way that the OEM key cannot be read or extracted clearly by the CM.
This is achieved by the fact that only the STM32 MCU can handle the OEM key.
In SFI for STM32H7RS, the OEM provisions its OEM key, using the Trusted Package Creator, in one Hardware Security Module (HSM). This must be done before providing the OEM key to the CM.
The STM32HSM-V2 HSM is used to secure the programming of STM32 products, and to avoid product counterfeiting at contract manufacturers' premises.
The Secure Firmware Install (SFI) feature allows secure downloading of customer firmware to STM32 products that embed a secure bootloader.
The advantages of this provisioning process are the

following
Only the STMicroelectronics STM32 microcontrollers can securely install the SFI image
The authenticity, integrity, and confidentiality of the SFI image content are ensured
When using the HSM, the number of STM32 chips to program can be counted.

SFI provisioning versus bootpaths STiRoT and OEMiROT

- SFI help provisioning Root Of Trust for the main 2 approaches:
  - using STiRoT
  - considering OEMiRoT



The STM32H7R/S is a hybrid solution, as applications are generally hosted in external memories.

The 64KB of embedded user Flash are used as bootflash to manage board specific aspects like the external memory management and the image loader.

Unlike STM32H5, there is no TZEN=enable/disable in Option bytes.

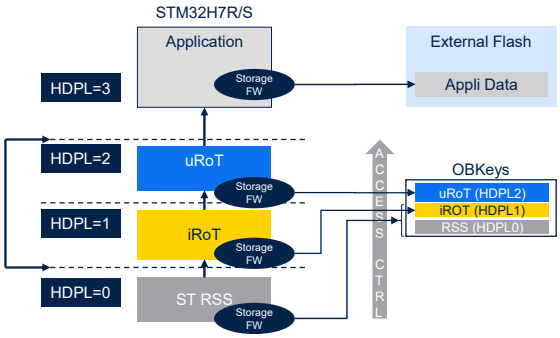
As soon as the product is no more in Open state, all boot goes through RSS.

This means that the RSS decides to jump to either ST-iROT, or OEM-iROT, or ST-Debug Authentication, or Boot Loader.

SFI helps provisioning the Root Of Trust for the main two approaches: ST-iRoT and OEM-iRoT.

STM32H7R/S HW Secure Data Storage

- Application
  - Data Storage firmware
    - Isolated in secure privileged
    - Provide services to non-privileged
- Key provisioning
  - During OEM product manufacturing (Provisioning state)
    - RSS-e-SFI when manufacturing not trusted
    - RSS-Lib when manufacturing is trusted

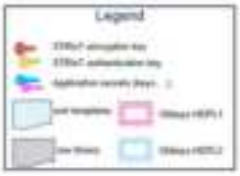


9

Hardware secure storage control improves the security, by isolating programs running at different privilege and security levels.
It also protects Option Byte Keys (OBKeys) against accesses and utilization by upper Hide Protection Levels (HDPLs).
Finally, hardware secure storage enables secure provisioning of firmware, by different OEMs.
Note that even if we are talking about key storage, any kind of data can be stored in the OBKeys.

SFI provisioning case of STM32H7S (STiRoT)

- In STM32H7R there is no ST-iRoT
  - The later is part of the STM32H7S only
- ST-iRoT configuration including OB keys is done via STM32 Trusted Package Creator
  - The ST-iRoT OBKeys configuration file (STiRoT_Config.obk) is generated using STM32 Trusted Package Creator with a template file listing all the different parameters (STiRoT_Config.xml) as inputs
  - The ST-iRoT configuration provides the possibility to:
    - Define the RAM and the external flash memory mapping: select the location and the size of each area
    - Configure the authentication and encryption keys
- Firmware binary: Code image generation using STM32Trusted Package Creator



10

Only the STM32H7S supports the ST-iROT.
ST-iRoT configuration including OB keys is done via STM32 Trusted Package Creator as follows.
The ST-iRoT OBKeys configuration file is generated using STM32 Trusted Package Creator with a template file listing all the different input parameters.
The ST-iRoT configuration provides the possibility to define the RAM and the external flash memory mapping, by selecting the location and the size of each area and also to configure the authentication and encryption keys.
The firmware binary code image generation is performed using STM32 Trusted Package Creator.

References

- For more details and additional information, refer to the following:
  - RM0477: STM32H7Rx/7Sx Arm®-based 32-bit MCUs Reference Manual
  - AN2606: STM32 microcontroller system memory boot mode
  - **AN4992**: Overview of Secure Firmware Install (SFI)
  - UM2237: STM32CubeProgrammer software description
  - UM2238: STM32 Trusted Package Creator software description
  - AN6088: How to use MCE for encryption/decryption on STM32 MCUs
  - AN6103: Guidelines for third-party programming on STM32H7Rx/7Sx MCUs (**NDA** is required to have access to this application note)
  - **SFI WIKI** pages for STM32H7RS: SFI for STM32H7Rx/7Sx - stm32mcu
  - X-Cube-SFI and X-Cube-RSSe: provides examples on how to use SFI

11

For more details, please refer to the following documents:
Application note AN2606 about the STM32 microcontroller system memory boot mode
Application note AN4992, which is an overview of Secure Firmware Install (SFI)
User manuals for the STM32 Cube Programmer and STM32 Trusted Package Creator
X-Cube-SFI // X-Cube-RSSe.

Thank you

Thank you for attending this presentation!
You can also refer to the following presentations on STM32H7RS security features:

- Security overview
- Cryptographic library (CRYPTOLIB)
- Memory Cipher Engine (MCE).