Hello, and welcome to this overview of security features present in the STM32U5.

New versus STM32L5

Key security features

Covered in this module

- Secure boot thanks to the unique boot entry and hide-protect area (HDP) features
- Improved resource isolation using TrustZone and privilege mode
- Enhanced life cycle management with Readout Protection (RDP)
  - Debug protection & optional password-based RDP regressions
- Dual-developer firmware distribution scheme
- On-the-fly decryption of encrypted images stored in external Flash memory

Covered in another training modules

- Enhanced secure storage
  - STM32U5-Security Enhanced key storage
- Cryptographic acceleration, including side-channel protection when manipulating secrets
  - STM32U5-Security Crypto
- Active tamper and protection against temperature, voltage and frequency attacks
  - STM32U5-Security Enhanced anti-tamper
- Secure firmware installation (SFI)
  - STM32U5-Security Secure firmware install
- Certification SESIP Level 3/ PSA level 3
  - STM32U5-Security Security Certification

life.augmented

2

The STM32U575/585 family of devices is designed with a comprehensive set of security features, some of which are based on standard Arm TrustZone technology. These security features simplify the process of evaluating IoT devices against security standards. They also significantly reduce the cost and complexity of software development for OEM and third-party developers, by facilitating re-use, improving interoperability, and minimizing API fragmentation.

This module describes the following key security features:

- Secure boot thanks to the unique boot entry and hide-protect area (HDP) features
- Improved resource isolation using TrustZone and privilege mode, extended to securable I/Os, memories

and peripherals
- Enhanced life cycle management with readout protection (RDP). It includes debug protection and optional password-based RDP regressions
- Dual-developer firmware distribution scheme using TrustZone, on-the-fly decryption and RDP0.5
- On-the-fly decryption of encrypted images stored in external Flash memory, with associated secure firmware installation

In other modules you find the following information:
- Enhanced secure storage
- Cryptographic acceleration, including side-channel protection when manipulating secrets
- Active tamper and protection against temperature, voltage and frequency attacks
- Secure Firmware installation
- Certification SESIP Level 3/ PSA level 3

New versus STM32L5

Secure boot key features

Boot modes when TrustZone is enabled

BOOT_ LOCK	RSSCMDR	Latched upon reset release				Boot area (must be in secure flash area)
		nBOOT0 FLASH_OPTR [27]	BOOT0 Pin PH3	nSWBOOT0 FLASH_OPTR [26]	FLASH secure boot address	
0	0	-	0	1	SECBOOTADD0	User defined (1) (secure only)
		-	1		*n/a*	RSS : 0x0FF8_0000
		1	-	0	SECBOOTADD0	User defined (1) (secure only)
		0	-		*n/a*	RSS : 0x0FF8_0000
	≠ 0	-	-	-	*n/a*	RSS : 0x0FF8_0000
1	-	-	-		SECBOOTADD0	User defined (1) (2) (secure only)

(1) Defined in secure user option byte SECBOOTADD0. Option byte default value is 0x0C00_0000
(2) Cannot be modified by the application anymore (unique boot entry enforcement)

Boot space versus RDP protection

RDP	Boot space (TZEN=1)
0	Any boot address
0.5	Allowed boot address: only in RSS or in secure Flash memory (0x0C00_0000-0x0C1F_FFFF) Wrong secure flash memory programming forces boot on RSS
1	
2	

life.augmented

3

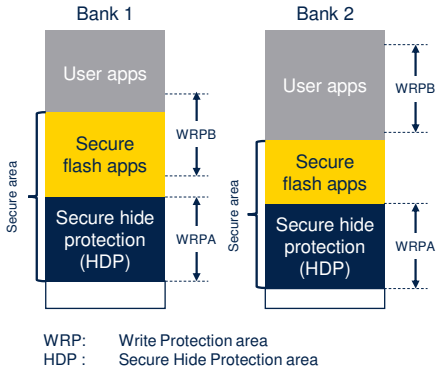
This slide summarizes the boot options when TrustZone is enabled:
- when BOOT_LOCK=1 the boot address is unique and defined by secure user option SECBOOTADD0, whatever the other parameters.
- when RSSCMDR is non null and BOOT_LOCK=0, boot in Root Security Services or RSS is performed.
- When RDP is greater than zero the boot code must be located in a secure area

When TrustZone is disabled only the center of the table is relevant, with non-secure NSBOOTADD0 replacing SECBOOTADD0, and non-secure NSBOOTADD1 replacing the RSS fixed address.

New versus STM32L5

Embedded flash key security features

- FLASH is configured in dual bank architecture only
- Like STM32L5 there are two write protection areas per bank (non-volatile configuration)
  - With non-volatile write protection locking U5 can emulate immutable memory (ROM)
    - L5 only relies on the HDP area
  - As write protection is unlocked following an RDP regression, the application must prevent such regressions to keep the area immutable (for example by provisioning a random OEM key password)
- Each 8 KB page of FLASH can be S/NS and/or P/NP (volatile configuration)



life.augmented

In embedded flash there are two write protection areas per bank, controlled by non-volatile configuration bits. These bits cannot be modified when the corresponding UNLOCK configuration bit is zero.
UNLOCK bits can be set only when regressing from RDP level 1 to level 0.
When TrustZone is enabled in the device, embedded flash features the following protections:

- One secure area per user flash bank defined with secure, non-volatile user option bytes. Default programming is all secure.
- One secure hide protection (HDP) area per bank, stickily hidden after boot by application
- 8 kilobytes user pages, on-the-fly defined by the

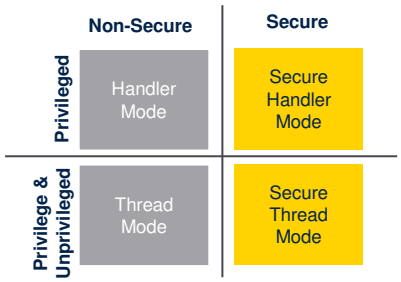
application as secure using volatile secure registers.
Default configuration out of reset is non-secure.

Additionally, each 8 kilo bytes user page can be defined on-the-fly as privileged only using volatile privileged registers.

When a page is defined secure, only secure applications can change this property.

Resource isolation improvements (1)

- To achieve SESIP level 3 – PSA level 3 certification applications can use four isolation states for system peripherals & memories
  - Secure/Privilege (S/P)
  - Secure/Non-Privilege (S/NP)
  - Non-Secure/Privilege (SN/P)
  - Non-Secure/ Non-Privilege (NS/NP)



life.augmented

5

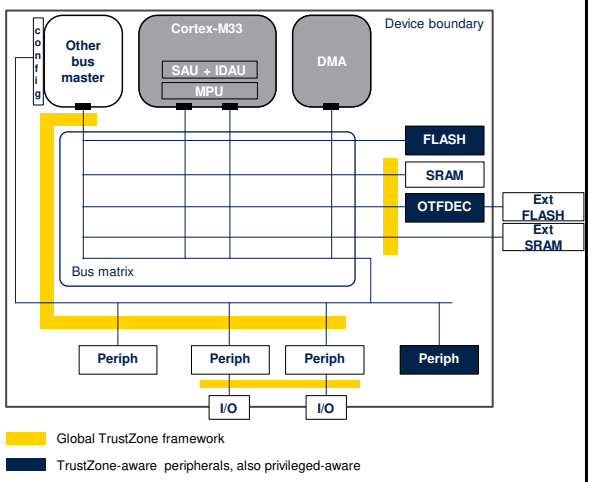
When TrustZone is enabled, the secure world can be used to protect critical code against intentional or unintentional tampering from the more exposed code running in the non-secure world.
Whether TrustZone is enabled or not, the Cortex privileged mode can be used to protect critical code or data against intentional or unintentional tampering from the more exposed unprivileged code.
These resource isolation features were instrumental to obtain the SESIP level 3 – PSA level 3 certification.
SESIP stands for Security Evaluation Standard for IoT Platforms (SESIP) and PSA stands for Platform Security Architecture.

New versus STM32L5

Resource isolation improvements (2)

Global TrustZone controller (GTZC)

- Securing peripherals (using TZSC)
  - Secure and privilege configuration per peripheral
- Securing memories (with MPCBB and TZSC)
  - Block-based secure and privilege configuration for internal SRAM
  - Secure and privilege configuration of external memories OCTOSPI / FSMC and backup SRAM sub-regions with watermarks
- Unlike the MPU, GTZC can protect legacy memories and peripherals against unprivileged transactions coming from masters other than the Cortex-M33
  - Secure and privilege configuration per legacy master
- Privileged isolation granularity in GTZC complements the coarse privileged isolation available in the Cortex Core (8x MPU regions)



life.augmented

On top of the Armv8-M TrustZone security extension in Cortex-M33, the devices embed complementary security features called the Global TrustZone Controller or GTZC that reinforce, in a flexible way, the isolation between respectively secure/non-secure worlds, and privileged/unprivileged worlds.
GTZC protects peripherals using registers in the TrustZone security controller or TZSC. It protects memories using the Memory Protection Controller - Block Based or MPCBB and the TZSC registers.
GTZC can protect against non-secure and optionally unprivileged transactions initiated by masters other than the Cortex-M33.
Note that some peripherals do not require GTZC to offer

secure or privilege protection, because they are natively TrustZone-aware and privileged-aware.

New versus STM32L5

Enhanced life cycle management (1)

RDP protection level		Debug	Lifecycle overview
Level 0	Device open	Secure[1] and non-secure	**Boot address must target a secure area when TZEN=1 (secure SRAM, secure Flash memory, RSS in system Flash memory)**
Level 0.5[2]	Device partially closed	Non-secure only	**Boot address must target a secure area (Secure user or system Flash memory)** ➢ Boot on SRAM is not permitted ➢ Access to non-secure Flash memory is allowed when debug is connected ➢ OEM1 key can be provisioned to prevent regression to level 0
Level 1	Device memories protected	Non-secure only (conditioned)	**Boot address must target the user or system Flash memory (secure if TZEN = 1)** ➢ Accesses to non-secure Flash memory, encrypted Flash memory[3], SRAM2 and backup registers are **not allowed** when debug is connected ➢ OEM1 key can be provisioned to prevent regression to level 0 ➢ OEM2 key can be provisioned to prevent regression to level 0.5
Level 2	Device closed	None (JTAG fuse)	**Boot address must target the user Flash memory (secure if TZEN = 1)** ➢ Option bytes are read-only, hence RDP level 2 cannot be changed, unless OEM2 key is provisioned and locked

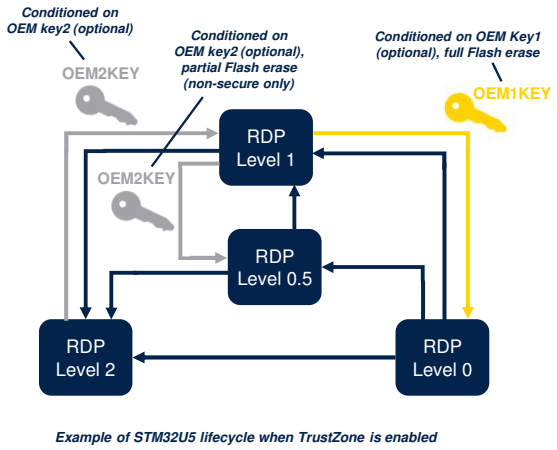
1. *Debug is not available when executing RSS code*
2. *Only applicable when TrustZone security is activated in the product (TZEN=1)*
3. *External Flash memory area decrypted on-the-fly with the OTFDEC*

life.augmented

The Device life cycle is managed by the readout protection option byte (RDP).
This RDP mechanism is a hardware feature that controls the access to the device debug, test and provisioned secrets, as summarized in this table.
The new features around unlocking keys are described in the following slide.

Enhanced life cycle management (1)

Password key-based RDP regressions (new feature not in STM32L5)



Example of STM32U5 lifecycle when TrustZone is enabled

- There are 2 password keys - **OEM1KEY**, **OEM2KEY**
  - Both 64-bit, write only, unreadable
  - If left un-programmed device works like a STM32L5
- OEM1KEY is used to manage the RDP level regression from **Level 1** to **Level 0**
- OEM2KEY is used to manage the RDP level regression from **Level 2** to **Level 1** and **Level 1** to **Level 0.5**
- To unlock and for RDP regression, the correct key must be shifted through JTAG / SWD pins **during reset**
- Unlocking the device with a password is possible only **once per power cycle**

life.augmented

8

Password key-based RDP regressions are available through the debug interface or via the system bootloader.

They are ideal for customers that do not want to irreversibly lock devices.

Two 64-bit keys are defined in embedded flash to independently protect secure (OEM1) and non-secure (OEM2) application codes, as shown.

One usage of this is described in the dual-developer firmware distribution scheme slide.

Note that unlocking the device with a password is only possible once per power cycle.

OEM1KEY can always be modified when RDP = 0. It can

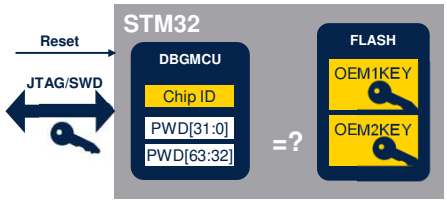
be changed when RDP = 0.5 or 1 if OEM1LOCK = 0.

OEM2KEY can always be modified when RDP = 0 or 0.5. It can be changed when RDP = 1 if OEM2LOCK = 0.

Enhanced life cycle management (2)

Associating password keys with a Chip ID (new feature not in STM32L5)

- A 32-bit device specific quantity can always be read though the JTAG/SWD port, except when the RDP legacy scheme applies (OEM2LOCK=0) and Level 2 is set (no more JTAG)
- OEM can use this Chip ID, together with secret master key(s), to provision device specific passwords in OEM1/2KEY option bytes
  - Once OEM key(s) are provisioned in FLASH, the OEM tools can read the chip ID and derive the expected key(s) to inject to unlock the device
- Getting the password will never compromise data confidentiality
  - It only allows to regress the RDP protection



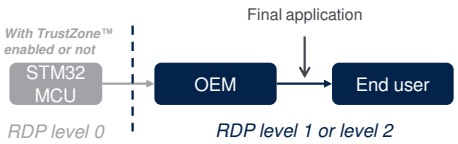
life.augmented

9

Through the debug interface a 32-bit device specific quantity can be read to compute device-specific passwords.
This method does not apply if RDP Level is 2 and OEM2LOCK=0.
Getting the password will never compromise data confidentiality.
It only allows to regress the RDP protection .

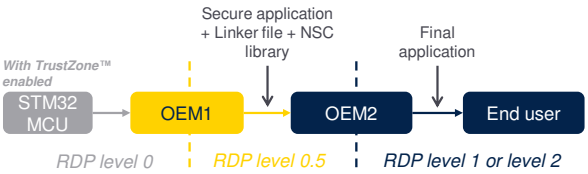
Dual-developer firmware distribution scheme

Single-developer approach



- A single developer (OEM) develops both secure & non-secure applications
  - Product firmware is protected using RDP level 1 or 2
  - There is no secure application when TrustZone is disabled

Dual-developer approach



- A first developer (OEM1) develops the secure application and its associated non-secure callable (NSC) library (.lib and .h)
- A second developer (OEM2) develops non-secure applications, using the linker file prepared by the OEM1
  - Secure application is protected using RDP level 0.5
  - Non-secure application is protected using RDP level 1 or RDP level 2

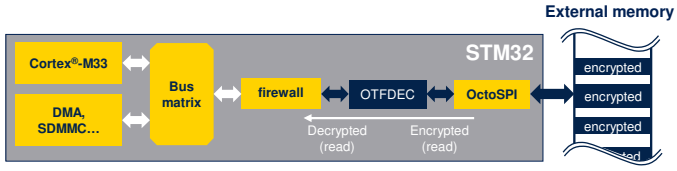
life.augmented

10

The device supports the STM32 dual-developer firmware distribution scheme:
- In the single-developer scheme, one OEM develops secure and non-secure applications. Both applications must be protected using RDP level 1 or RDP level 2.
- In the dual-developer scheme, the first OEM develops the secure application, its associated non-secure callable library, and provides a predefined linker file to the second OEM that develops the non-secure application.
In the dual-developer scheme the secure application must be protected using RDP level 0.5 after installation. The final non-secure application must be protected using RDP level 1 or RDP level 2.

On–The-Fly Decryption Engine (OTFDEC) features

- Decrypts on-the-fly read-only information encrypted in external SPI Flash
  - Four regions can be defined, each with a dedicated key and public diversification data
- Uses standard AES-128 in counter mode, with an optional enhanced encryption option (instructions only)
- Write-only key registers, write protection until next reset (KEYLOCK & CONFIGLOCK)
- Global security mechanisms
  - Key erase in case of intrusion, RDP regression or MODE change
  - TrustZone-aware peripheral (register writes always secure when TZEN=1)
  - Privileged-only accesses when PRIV bit is set in OTFDEC_PRIVCFGR
- Encryption mode (Secure only)

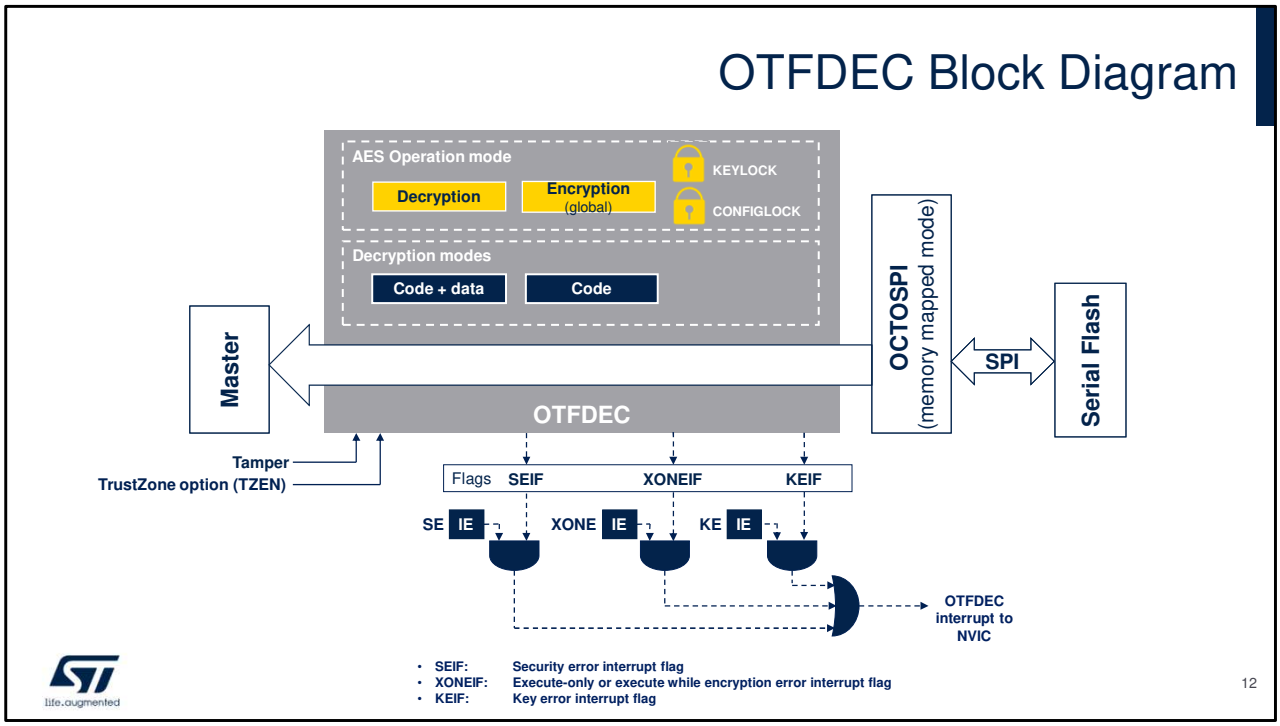


life.augmented

11

The OTFDEC module decrypts on-the-fly read-only information encrypted in external SPI Flash.
AES 128-bit cipher in counter mode is used to achieve the lowest possible latency.
Four independent and non-overlapping encrypted regions can be defined. For each region an additional layer of protection can be added on top of the standard AES encryption algorithm, requiring the encryption to be done on-chip.
When such enhanced protection is selected only instructions can be stored in the region.
OTFDEC also supports an encryption mode, available when no decryption is on-going.
All key registers are write-only, and automatically erased in

the case of tampers or RDP regression.
OTFDEC is a TrustZone-aware peripheral.
All writes to its registers must be secure when security is activated in the product (TZEN=1).
When PRIV bit is set in OTFDEC, only privileged accesses are granted when accessing most of the OTFDEC registers.



OTFDEC analyzes all AHB read transfers on the associated AHB bus. If the read request is within one of the four regions programmed the control logic triggers a keystream computation based on the AES algorithm in counter mode.

This keystream is then used to decrypt the data present on-the-fly, in the read transfer from the OCTOSPI AHB master.

Any access outside the enabled OTFDEC regions belongs to a non-encrypted region.

As OTFDEC is used in conjunction with OCTOSPI it is mandatory to access the flash memory using the memory mapped mode of the flash controller.

The OTFDEC can assert an interrupt to the NVIC for three

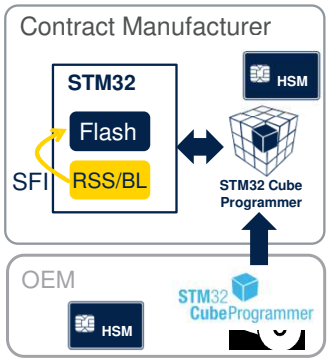
possible causes:

- Security error
- Key error
- Execute-only or execute while encryption error

Each of these errors has a dedicated flag and an interrupt enable bit.

Secure firmware install with OTFDEC

- With Secure Firmware Install (SFI) secure and counted installation of OEM firmware in untrusted production environments (such as an OEM contract manufacturer) is possible
- When external Flash memory is targeted by SFI, the OEM firmware is encrypted with a dedicated AES key
- OTFDEC can be used to encrypt this external firmware, for example with a unique device key
  - This option is mandatory when enhanced encryption is selected for the region
- Refer to AN4992 for more details on SFI



life.augmented

13

Secure firmware install (SFI) is a global solution for this STM32 series of microcontrollers, allowing secure and counted installation of OEM firmware in untrusted production environments (such as an OEM contract manufacturer).
When external Flash memory is targeted by SFI, the OEM firmware code must be encrypted with a dedicated AES key.
This key can be:

- Common to a family of products, with OEM tools performing the encryption
- Unique per device, with the firmware encrypted inside the device

On-chip encryption is mandatory when OTFDEC

enhanced encryption is selected.
For more information, please refer to application note AN4992 for secure firmware install (SFI) solutions.

Thank you

life.augmented

Thank you for having attended this presentation!
You can now refer to the presentations that detail the operation of the STM32U5's security modules:

- Symmetric cryptography.
- Asymmetric cryptography.
- Hash and random number generation.
- Enhanced anti-tamper.
- Enhanced key storage.