SHE API

Generated by Doxygen 1.8.11

Contents

1 Main Page

2 Module Index

2.1 Modules

Here is a list of all modules:

3 File Index

3.1 File List

Here is a list of all files with brief descriptions:

4 Module Documentation

4.1 She_api

SHE feature API.

Macros

- #define SHE_MAC_SIZE 16
- #define SHE_MAC_VERIFICATION_SUCCESS 0
- #define SHE_MAC_VERIFICATION_FAILED 1
- #define SHE_AES_BLOCK_SIZE_128 16

Enumerations

- enum she_err_t {
  ERC_NO_ERROR = 0x0, ERC_SEQUENCE_ERROR = 0x1, ERC_KEY_NOT_AVAILABLE = 0x2, ERC_KEY_INVALID = 0x3,
  ERC_KEY_EMPTY = 0x4, ERC_NO_SECURE_BOOT = 0x5, ERC_KEY_WRITE_PROTECTED = 0x6, ERC_KEY_UPDATE_ERROR = 0x7,
  ERC_RNG_SEED = 0x8, ERC_NO_DEBUGGING = 0x9, ERC_BUSY = 0xA, ERC_MEMORY_FAILURE = 0xB,
  ERC_GENERAL_ERROR = 0xC }

  Error codes returned by SHE functions.

Functions

- struct she_hdl_s * she_open_session (void)
- void she_close_session (struct she_hdl_s *hdl)
- she_err_t she_cmd_generate_mac (struct she_hdl_s *hdl, uint8_t key_id, uint32_t message_length, uint8_t *message, uint8_t *mac)
- she_err_t she_cmd_verify_mac (struct she_hdl_s *hdl, uint8_t key_id, uint32_t message_length, uint8_t *message, uint8_t *mac, uint8_t mac_length, uint8_t *verification_status)
- she_err_t she_cmd_enc_cbc (struct she_hdl_s *hdl, uint8_t key_id, uint32_t data_length, uint8_t *iv, uint8_t *plaintext, uint8_t *ciphertext)
- she_err_t she_cmd_dec_cbc (struct she_hdl_s *hdl, uint8_t key_id, uint32_t data_length, uint8_t *iv, uint8_t *ciphertext, uint8_t *plaintext)
- she_err_t she_cmd_enc_ecb (struct she_hdl_s *hdl, uint8_t key_id, uint8_t *plaintext, uint8_t *ciphertext)
- she_err_t she_cmd_dec_ecb (struct she_hdl_s *hdl, uint8_t key_id, uint8_t *ciphertext, uint8_t *plaintext)
- she_err_t she_cmd_load_key (struct she_hdl_s *hdl)

4.1.1 Detailed Description

SHE feature API.

4.1.2 Macro Definition Documentation

4.1.2.1 #define SHE_AES_BLOCK_SIZE_128 16

size in bytes of a 128bits CBC bloc

4.1.2.2 #define SHE_MAC_SIZE 16

size of the MAC generated is 128bits.

4.1.2.3 #define SHE_MAC_VERIFICATION_FAILED 1

indication of mac verification failure

4.1.2.4 #define SHE_MAC_VERIFICATION_SUCCESS 0

indication of mac verification success

4.1.3 Enumeration Type Documentation

4.1.3.1 enum she_err_t

Error codes returned by SHE functions.

Enumerator

ERC_NO_ERROR Success.
ERC_SEQUENCE_ERROR Invalid sequence of commands.
ERC_KEY_NOT_AVAILABLE Key is locked.
ERC_KEY_INVALID Key not allowed for the given operation.
ERC_KEY_EMPTY Key has not beed initialized yet.
ERC_NO_SECURE_BOOT Conditions for a secure boot process are not met.
ERC_KEY_WRITE_PROTECTED Memory slot for this key has been write-protected.
ERC_KEY_UPDATE_ERROR Key update did not succeed due to errors in verification of the messages.
ERC_RNG_SEED The seed has not been initialized.
ERC_NO_DEBUGGING Internal debugging is not possible.
ERC_BUSY A function of SHE is called while another function is still processing.
ERC_MEMORY_FAILURE Memory error (e.g. flipped bits)
ERC_GENERAL_ERROR Error not covered by other codes occured.

4.1.4 Function Documentation

4.1.4.1 void she_close_session (struct she_hdl_s * *hdl*)

Terminate a previously opened SHE session

Parameters

hdl	pointer to the session handler to be closed.

4.1.4.2 she_err_t she_cmd_dec_cbc (struct she_hdl_s * *hdl*, uint8_t *key_id*, uint32_t *data_length*, uint8_t * *iv*, uint8_t * *ciphertext*, uint8_t * *plaintext*)

CBC decryption of a given ciphertext with the key identified by key_id.

Parameters

hdl	pointer to the SHE session handler
key_id	identifier of the key to be used for the operation
data_length	lenght in bytes of the plaintext and the cyphertext. Must be a multiple of 128bits.
iv	pointer to the 128bits IV to use for the decryption.
ciphertext	pointer to ciphertext to be decrypted.
plaintext	pointer to the plaintext output area.

Returns

error code

4.1.4.3 she_err_t she_cmd_dec_ecb (struct she_hdl_s * *hdl*, uint8_t *key_id*, uint8_t * *ciphertext*, uint8_t * *plaintext*)

ECB decryption of a given ciphertext with the key identified by key_id.

Parameters

hdl	pointer to the SHE session handler
key_id	identifier of the key to be used for the operation
ciphertext	pointer to 128bits ciphertext to be decrypted.
plaintext	pointer to the plaintext output area (128bits).

Returns

error code

4.1.4.4 she_err_t she_cmd_enc_cbc (struct she_hdl_s * *hdl*, uint8_t *key_id*, uint32_t *data_length*, uint8_t * *iv*, uint8_t * *plaintext*, uint8_t * *ciphertext*)

CBC encryption of a given plaintext with the key identified by key_id.

Parameters

hdl	pointer to the SHE session handler
key_id	identifier of the key to be used for the operation
data_length	lenght in bytes of the plaintext and the cyphertext. Must be a multiple of 128bits.
iv	pointer to the 128bits IV to use for the encryption.
plaintext	pointer to the message to be encrypted.
ciphertext	pointer to ciphertext output area.

Returns

error code

4.1.4.5 she_err_t she_cmd_enc_ecb (struct she_hdl_s * *hdl*, uint8_t *key_id*, uint8_t * *plaintext*, uint8_t * *ciphertext*)

ECB encryption of a given plaintext with the key identified by key_id.

Parameters

hdl	pointer to the SHE session handler
key_id	identifier of the key to be used for the operation
plaintext	pointer to the 128bits message to be encrypted.
ciphertext	pointer to ciphertext output area (128bits).

Returns

error code

4.1.4.6 she_err_t she_cmd_generate_mac (struct she_hdl_s * *hdl*, uint8_t *key_id*, uint32_t *message_length*, uint8_t * *message*, uint8_t * *mac*)

Generates a MAC of a given message with the help of a key identified by key_id.

Parameters

hdl	pointer to the SHE session handler
key_id	identifier of the key to be used for the operation
message_length	lenght in bytes of the input message
message	pointer to the message to be processed
mac	pointer to where the output MAC should be written (128bits should be allocated there)

Returns

error code

4.1.4.7 she_err_t she_cmd_load_key (struct she_hdl_s * *hdl*)

Temporary: Entry point to test NVM storage. Will be modified to support all parameters really neededd by load key command.

4.1.4.8 she_err_t she_cmd_verify_mac (struct she_hdl_s * *hdl*, uint8_t *key_id*, uint32_t *message_length*, uint8_t * *message*, uint8_t * *mac*, uint8_t *mac_length*, uint8_t * *verification_status*)

Verifies the MAC of a given message with the help of a key identified by key_id.

Parameters

hdl	pointer to the SHE session handler

Parameters

key_id	identifier of the key to be used for the operation
message_length	lenght in bytes of the input message
message	pointer to the message to be processed
mac	pointer to the MAC to be compared (implicitely 128 bits)
mac_length	number of bytes to compare (must be at least 4)
verification_status	pointer to where write the result of the MAC comparison

Returns

error code

4.1.4.9 struct she_hdl_s* she_open_session (void)

Initiate a SHE session. The returned session handle pointer is typed with the transparent struct "she_hdl_s". The user doesn't need to know or to access the fields of this struct. It only needs to store this pointer and pass it to every calls to other APIs within the same SHE session.

Returns

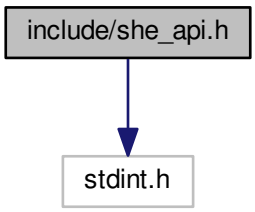
pointer to the session handle.

5 File Documentation

5.1 include/she_api.h File Reference

```
#include <stdint.h>
```

Include dependency graph for she_api.h:



Macros

- #define SHE_MAC_SIZE 16
- #define SHE_MAC_VERIFICATION_SUCCESS 0
- #define SHE_MAC_VERIFICATION_FAILED 1
- #define SHE_AES_BLOCK_SIZE_128 16

Enumerations

- enum she_err_t {
  ERC_NO_ERROR = 0x0, ERC_SEQUENCE_ERROR = 0x1, ERC_KEY_NOT_AVAILABLE = 0x2, ERC_↩
  KEY_INVALID = 0x3,
  ERC_KEY_EMPTY = 0x4, ERC_NO_SECURE_BOOT = 0x5, ERC_KEY_WRITE_PROTECTED = 0x6, E↩
  RC_KEY_UPDATE_ERROR = 0x7,
  ERC_RNG_SEED = 0x8, ERC_NO_DEBUGGING = 0x9, ERC_BUSY = 0xA, ERC_MEMORY_FAILURE =
  0xB,
  ERC_GENERAL_ERROR = 0xC }

  Error codes returned by SHE functions.

Functions

- struct she_hdl_s ∗ she_open_session (void)
- void she_close_session (struct she_hdl_s ∗hdl)
- she_err_t she_cmd_generate_mac (struct she_hdl_s ∗hdl, uint8_t key_id, uint32_t message_length, uint8_t ∗message, uint8_t ∗mac)
- she_err_t she_cmd_verify_mac (struct she_hdl_s ∗hdl, uint8_t key_id, uint32_t message_length, uint8_↩ t ∗message, uint8_t ∗mac, uint8_t mac_length, uint8_t ∗verification_status)
- she_err_t she_cmd_enc_cbc (struct she_hdl_s ∗hdl, uint8_t key_id, uint32_t data_length, uint8_t ∗iv, uint8↩ _t ∗plaintext, uint8_t ∗ciphertext)
- she_err_t she_cmd_dec_cbc (struct she_hdl_s ∗hdl, uint8_t key_id, uint32_t data_length, uint8_t ∗iv, uint8↩ _t ∗ciphertext, uint8_t ∗plaintext)
- she_err_t she_cmd_enc_ecb (struct she_hdl_s ∗hdl, uint8_t key_id, uint8_t ∗plaintext, uint8_t ∗ciphertext)
- she_err_t she_cmd_dec_ecb (struct she_hdl_s ∗hdl, uint8_t key_id, uint8_t ∗ciphertext, uint8_t ∗plaintext)
- she_err_t she_cmd_load_key (struct she_hdl_s ∗hdl)

5.2 include/she_storage.h File Reference

Functions

- struct she_storage_context ∗ she_storage_init (void)
- void she_storage_terminate (struct she_storage_context ∗ctx)

5.2.1 Function Documentation

5.2.1.1 struct she_storage_context∗ she_storage_init (void)

Initialize SHE storage manager.

Returns

pointer to the storage context

5.2.1.2 void she_storage_terminate (struct she_storage_context ∗ *ctx*)

terminates the SHE storage manager.

Parameters

ctx	pointer to the context of the storage manager to be closed.

Index