APPLICATION NOTE

CryptoAuthentication Personalization Guide

ATSHA204A and ATECC508A

Introduction

Before using the Atmel® CryptoAuthentication™ ATSHA204A and ATECC508A devices (crypto devices), there are some initialization processes that need to be performed first. The initialization processes consist of personalizing the device and then locking the device.

In the personalization step, the device behavior, the data slot behavior, and the data itself is being configured as desired. After the personalization process is performed, the device needs to be locked to prevent any further modification to data. Details of the initialization processes are described within this document.

Features

The initialization process consists of four basic steps:

- **Personalize Configuration Zone** – Personalize device configuration such as I2C_Address, OTPMode, SelectorMode, SlotConfig, UseFlag, LastKeyUse, and Selector Value.
- **Lock Configuration Zone** – Lock Configuration zone after the Configuration zone has been personalized. Atmel recommends including CRC checksum in the lock process to ensure that the device has been personalized as desired.

  Configuration zone must be locked to enable the personalization of Data/OTP zones. Prior to locking Configuration zone, Data/OTP zone cannot be read nor written at all.
- **Personalize Data/OTP Zones** – Personalize Data zone to store data such as keys, calibration data, model number, etc.

  Personalize OTP zone to store fixed data such as model numbers, calibration information, manufacturing history, or other data that should never change.
- **Lock Data/OTP Zones** – Lock Data/OTP zones after writing Data and OTP zones. Atmel recommends including CRC checksum in the lock process to ensure the data written are as desired.

1 Personalize Configuration Zone

There are some bytes that should be configured before using the crypto device. These bytes are used to control the access permission information for each slot of the data memory and to personalize the device behavior itself.

The details of these bytes are described below. For more information, please refer to latest ATSHA204A or ATECC508A datasheet.

- **I2C_Address** — I2C_Address is represented by byte 16 of the Configuration zone. It holds `0xC8` as a default value for ATSHA204A and `0xC0` for ATECC508A. This byte is mainly used to identify the device in I^2C communication. This byte can also be used to control the input level for the crypto devices.
- **OTPMode** — This OTPMode is represented by byte 17 of the Configuration zone. It holds `0x55` as default value. This byte is used to control the permission level of OTP zones. OTP zone can be used as additional message body for generating `MAC`, `HMAC`, or `GenDig` response, according to the command mode.
- **SelectorMode** — This bit is used to control the write permission to Selector byte. The SelectorMode is represented by bit 0 of byte 19 within the Configuration Zone. If this bit is cleared, the Selector byte can always be modified with `UpdateExtra` command. If this bit is set, the Selector byte can only be written if it currently has a value of zero.
- **Slot Configuration** — Slot Configuration bytes are represented by byte 20 – 51. Each slot uses two bytes to determine the slot behavior, byte 20 and byte 21 are the configuration bytes for key slot 1, byte 22 and byte 23 which are the configuration bytes for key slot 2, and so on.

  Note: The even bytes are LSBytes – it holds bits 0:7, while the odd bytes are MSBytes — it holds bits 8:15. For example, slot 1 configuration bytes are byte 20 and byte 21, bits 0:7 are hold by byte 20, while bits 8:15 are hold by byte 21.

  See the below table for details of the 2-byte configurations of each slot.

Table 1-1. 2-byte Slot Configurations

Bit	Device	SlotConfig	Description
0 → 3	Both	ReadKey	Determine which KeyID will be used to generate the encryption key to encrypt the data being read from the corresponding slot.
4	ATSHA204A	CheckOnly	Determine whether the corresponding slot is used for `CheckMac` command only or can be used for all crypto commands.
	ATECC508A	NoMac	Determine whether the corresponding slot cannot be used by `MAC` or `HMAC` command or can be used for all crypto commands.
5	ATSHA204A	SingleUse	Determine whether the usage of the corresponding slot is limited or not. This limitation only applied to slot 0 thru 7 and 15. Must be zero for keys in Slots 8 thru 14.
	ATECC508A	LimitedUse	Determine whether the usage of Slot 15 is limited or not. If this bit is set on Slot 0 thru 14, then any use of the keys will cause counter[0] to increment automatically prior to the operation being performed.
6	Both	EncryptRead	Determine that reading from the corresponding slot must be encrypted or not. EncryptRead is represented by bit 6 of the slot configuration.
7	Both	IsSecret	Determine that the corresponding slot is secret or not.
8 → 11	Both	WriteKey	Determine which KeyID will be used to validate and encrypt data written to the corresponding slot.
12 → 15	Both	WriteConfig	Determine the modification ability of the corresponding slot. There are two ways to modify the data in the slot; by using `Write` command and by using `DeriveKey` command. The WriteConfig control the ability of these two commands to modify the data.

- **UseFlag / Counter[0:1]** — For ATSHA204A, UseFlag is represented by even byte from byte 52 to 67. Byte 52 corresponds to Key0, 54 to Key1, and so on. These bytes are used to indicate how many times a key may be used. This limitation is only applied to key slot 0 thru 7. By default, this byte is set to `0xFF`. Each time the key is used, a UseFlag bit changed from one to zero and it starts from the most significant bit to the least significant bit. By default, a key may be used eight times before it must be refreshed by using a `Write` or `DeriveKey` command; however, these limitations can be reduced by clearing some of the most significant bit, into `0x7F` (seven uses), `0x3F` (six uses), `0x1F` (five uses), and so on. Atmel recommends that the key to be used a single time only, with the other chance of uses providing a safety margin for errors.

  For ATECC508A, byte 52 to 67 of the Configuration zone are used as Monotonic Counters. Counter[0] can optionally be connected to key 0 thru 14 via the SlotConfig.LimitedUse bit. When Counter[0] is attached to a key, the counter will be incremented with each use of the key until the counter has reached its maximum value at which point use of the key will no longer be permitted. The number of legal uses for a key can be controlled by initializing the Counter[0] to a non-zero value at configuration time. Contact Atmel for details. Both Counter[0] and Counter[1] can be incremented via the `Counter` command.
- **LastKeyUse** — LastKeyUse is represented by bytes 68 – 83. The default value of these bytes is `0xFF`. These bytes act similarly like UseFlag, but they are only applied to Key15. Each time Key15 is used, the same mechanism as UseFlag is applied here. Therefore, this key can be limited up to 128 uses. The user can reduce the use limitation by setting these bytes the exact same way as setting the UseFlag byte. The total number of bits set to one indicates the number of usage limitations.

  After all the LastKeyUse reached `0x00`, key15 is permanently disabled. There is no mechanism reset the LastKeyUse bytes.
- **Selector** — Selector is represented by byte 85. The default value of this byte is `0x00`. This byte is used to select which chip will remain in active mode after the execution of pause command. This byte cannot be modified by using normal write command; instead it can only be updated using `UpdateExtra` command.
- **Key Configuration** — For ATECC508A, Key Configuration bytes are represented by byte 96 – 127. The 16 KeyConfig elements are used in addition to SlotConfig to restrict the actions that can be performed using information stored in a particular slot. The KeyConfig element is interpreted according to the table below when the Data zone is locked. When the Data zone is unlocked, these restrictions do not apply, with the exception that slots configured to contain private keys can be written only with the `PrivWrite` command. KeyType, which is represented by bit 2 – 4 of KeyConfig, must be set on every slot that does not contain an ECC key.

2 Lock Configuration Zone

After the crypto device is configured, the next step is to lock Configuration zone. Prior to locking Configuration zone, neither read nor write is permitted to the Data/OTP zones; therefore, the Configuration zone must be locked before personalizing Data and OTP zones.

Atmel recommends using CRC checksum in the lock process to ensure the device has been configured as desired. The crypto device uses CRC-16 algorithm to generate a summary digest of the designated zones. For Configuration zone, the CRC is calculated over all bytes (88 bytes in ATSHA204A; 128 bytes in ATECC508A) of the Configuration zone. If the CRC does not match, an error is returned from the device, indicating there is data mismatch. If there is data mismatch, the personalization process needs to be repeated to ensure every personalization is as desired.

3 Personalize Data/OTP Zones

ATSHA204A Data zone consists of 512 bytes split into 16 equal-sized slots. Whereas, the ATECC508A Data zone consists of 1208 bytes split into 16 slots of varying sizes. Each slot access restrictions are individually programmable and can be used to store secret key, calibration data, model number, or other information related to the item to which the crypto device is attached. Additionally, ATECC508A slots can be used to store Private Key, Public Key, Certificate, and/or Signature. While all slots can be used to store Private Key or User Data, only Slots 8 thru 15 are large enough to store an ECC Public Key or ECDSA Certificate/Signature.

To be used for Asymmetric Authentication, the ATECC508A support internal generation of Private and Public Key pair via `GenKey` command. If the Private and Public Key pair is externally generated, the Private Key can also be stored using `PrivWrite` Command. Other data aside from the Private Key can be stored using `Write` command.

OTP zone consists of 64 bytes of one-time programmable (OTP) bits. The OTP zones can be used to store fixed data such as, model numbers, calibration information, manufacturing history, or other data that should never change. These bytes can freely be written after the Configuration zone has been locked, but prior to Data/OTP zones locked.

4 Lock Data/OTP Zones

Upon completion of any writes, the Data and OTP sections should be locked. It is important that the Data and OTP sections be locked prior to release of the system containing the device into the field. Failure to lock these zones may permit modification of any secret keys and may lead to other security problems.

Atmel recommends using CRC checksum in the lock process to ensure the data written are as desired. The crypto device uses CRC-16 algorithm to generate a summary digest of the designated zones. For the Data and OTP zones locked, the contents are concatenated in the order to create the input to the CRC algorithm. If the CRC does not match, an error is returned from the device, indicated that there is data mismatch; therefore the personalization process needs to be repeated.

5 Personalization Example

This section gives an example of the desired device personalization and the corresponding bits setting for the device.

5.1 Accessory Authentication Use Case

Two crypto devices are used in an accessory authentication, such as battery authentication. In this case, the crypto device is embedded in a mobile device as Host and in the battery as Client. The symmetric diversified key scheme is also utilized in this example case. The communication protocol used between MCU and the crypto device is I^2C protocol.

5.1.1 Host Crypto Device Personalization

The personalization of the Host for this example case is described as follows.

- I2C_Address — Set this byte as `0xAA`. Another value can be used as long as it is different from the Client crypto device I2C_Address.
- SlotConfig for the Master Key Slot. (i.e. slot 0) — This key is used to generate diversified key, which is used in the authentication process; therefore, the key must be configured to be “No Read or Write permitted” on this slot.

  The SlotConfig for this slot is `0x81 80`. See the below table for more details.

Table 5-1. SlotConfig for the Master Key Slot

SlotConfig	Comments
ReadKey	Can be set to any value other than zero to avoid the CheckMac copy operation, i.e. `0x1`.
CheckOnly / NoMac	Must be set to zero to enable all crypto functions on this slot.
SingleUse / LimitedUse	Must be set to zero to disable the limited usage.
EncryptRead	Must be set to zero to disable read in encryption mode.
IsSecret	Must be set to one to disable read in clear text mode.
WriteKey	Can be set to any value, i.e. `0x0`.
WriteConfig	Must be set to disable `Write` and `DeriveKey` command, i.e. `0x8`.
KeyType	Must be set to seven as this slot is not used to store ECC key.

- SlotConfig for the Diversified Key Slot (i.e. slot 1) — Used to check the authenticity of the accessory. The key must be configured to be No Read, while the content can only be modified by deriving from Master Key.

  The SlotConfig for this slot is `0x91 30`.

Table 5-2. SlotConfig for the Diversified Key Slot

SlotConfig	Comments
ReadKey	Can be set to any value other than zero to avoid the CheckMac copy operation, i.e. `0x1`.
CheckOnly / NoMac	Must be set to one, since it is only used for performing authentication with the accessory.
SingleUse / LimitedUse	Must be set to zero to disable the limited usage.
EncryptRead	Must be set to zero to disable read in encryption mode.
IsSecret	Must be set to one to disable read in clear text mode.
WriteKey	Must be set to Master Key, in this case, `0x0`.
WriteConfig	Can be set to `0x3` or `0xB` depends on the authorization is required or not, in this example, it is set to `0x3`.
KeyType	Must be set to seven as this slot is not used to store ECC key.

- Other settings can be left as default.

5.1.2 Client Crypto Device Personalization

The personalization of the client for this example case is shown below.

- I2C_Address — Set this byte as `0xBB`. Other value can be used as long as it is different from the Host crypto device I2C_address.
- SlotConfig for the Diversified Key Slot (i.e. slot 0) — Used to generate response for the authentication process. The key must be configured to be “No Read or Write permitted” on this slot.

  The SlotConfig for this slot is `0x81 80`.

Table 5-3. SlotConfig for the Diversified Key Slot

SlotConfig	Comments
ReadKey	Can be set to any value other than zero to avoid the CheckMac copy operation, i.e. `0x1`.
CheckOnly / NoMac	Must be set to zero to enable all crypto functions on this slot.
SingleUse / LimitedUse	Must be set to zero to disable the limited usage.
EncryptRead	Must be set to zero to disable read in encryption mode.
IsSecret	Must be set to one to disable read in clear text mode.
WriteKey	Can be set to any value, i.e. `0x0`.
WriteConfig	Must be set to disable `Write` and `DeriveKey` command, i.e. `0x8`.
KeyType	Must be set to seven as this slot is not used to store ECC key.

- Other settings can be left as default.

5.2 Consumable Authentication Use Case

In this case, the crypto device is used to authenticate a consumable, track the consumable uses, and also limit the consumable uses. The crypto device is embedded in the Host and also in the consumable. In this example, the communication protocol being used is SWI interface.

5.2.1 Host Crypto Device Personalization

The personalization of the Host for this example case is described below.

- **SelectorMode** — Can be set to any value other than zero to prevent further modification of Selector byte.
- **Slotconfig for the Authentication Key Slot (i.e. slot 0)** — Used to check the authenticity of the consumable. The key must be configured to be “No Read or Write permitted” on this slot.

  The SlotConfig for this slot is `0x91 80`. See the below table for details.

Table 5-4. Slotconfig for the Authentication Key Slot

SlotConfig	Comments
ReadKey	Can be set to any value other than zero to avoid the CheckMac copy operation, i.e. `0x1`.
CheckOnly / NoMac	Must be set to one to enable only `CheckMac` command.
SingleUse / LimitedUse	Must be set to zero to disable the limited usage.
EncryptRead	Must be set to zero to disable read in encryption mode.
IsSecret	Must be set to one to disable read in clear text mode.
WriteKey	Can be set to any value, i.e. 0x0.
WriteConfig	Must be set to disable `Write` and `DeriveKey` command, i.e. `0x8`.
KeyType	Must be set to seven as this slot is not used to store ECC key.

- **Selector** — Set this byte as `0xAA`. Other value can be used as long as it is different from the Client crypto device Selector byte. Value `0x00` is prohibited to prevent further modification on this byte.
- Other settings can be left as default.

5.2.2 Client Crypto Device Personalization

The personalization of the Host for this example case is described below.

- **SelectorMode** — Can be set to any value other than zero to prevent further modification of Selector byte.
- **SlotConfig for the Authentication Key Slot** — Must use Slot15. Used to generate the response for the authentication process. The key must be configured to be “No Read or Write permitted” on this slot and also configured to be used for a limited use only, i.e. 64 uses.

  The SlotConfig for this slot is `0xA1 80`. See the following table for details.

Table 5-5. SlotConfig for the Authentication Key Slot

SlotConfig	Comments
ReadKey	Can be set to any value other than zero to avoid the CheckMac copy operation, i.e. `0x1`.
CheckOnly / NoMac	Must be set to zero to enable all crypto functions on this slot.
SingleUse / LimitedUse	Must be set to one to enable the limited usage.
EncryptRead	Must be set to zero to disable read in encryption mode.
IsSecret	Must be set to one to disable read in clear text mode.
WriteKey	Can be set to any value, i.e. `0x0`.
WriteConfig	Must be set to disable `Write` and `DeriveKey` command, i.e. `0x8`.
KeyType	Must be set to seven as this slot is not used to store ECC key.

- **LastKeyUse** — Set these bytes to `0x00 00 00 … 00 FF FF FF FF FF FF FF FF`. These bytes limit the usage of KeyID 15 to 64 uses.
- **Selector** — Set this byte as `0xBB`. Other value can be used as long as it is different from the Host crypto device Selector byte. Value `0x00` is prohibited to prevent further modification on this byte.
- Other settings can be left as default.

6 Revision History

Doc Rev.	Date	Comments
8845C	07/2015	Added the ATSHA204A and ATECC508A devices.
8845B	05/2013	Simplified the document and added another example.
8845A	11/2011	Initial document release.

Atmel Corporation 1600 Technology Drive, San Jose, CA 95110 USA **T:** (+1)(408) 441.0311 **F:** (+1)(408) 436.4200 | **www.atmel.com**