AUTOSAR CryIf User Manual

Document Change History				
Date (YYYY-MM-DD)	Ver.	Editor	Chap	Content
2021-01-15	1.0.0.0	JaeHyun Lim	All	▪ Initial Version
2021-03-20	1.0.1.0	TamTV6	4.3	▪ Update change log
2021-11-12	1.0.2.0	TamTV6	All	▪ Applying change of company name
2022-07-01	1.0.2.1	DienTC1	4.3	▪ Update Copyright and company name
2022-07-20	1.0.2.2	DienTC1	4.3	▪ Fix TM 100% coverage
2022-08-23	1.0.3.0	DienTC1	4.3	▪ Fix UNECE violations coding security
2022-12-07	1.0.4.0	DienTC1	4.3	▪ Correct implementation for API CryIf_KeyCopy(). ▪ Fix wrong generation in CryIf_Cfg.h in the case CryptoDriver module description vendorApiInfix is empty.
2023-03-03	1.0.5.0	PhuocLH9	4.3	▪ Update Change Log

Table of Contents

Table of Figures

1 Overview

It is written based on AUTOSAR standard SRS / SWS. If more detailed functional explanation is needed when using the module, see the Reference Manual. The interpretation of setting related category is as follows:

- Changeable (C): Items that can be set by the user
- Fixed (F): Items that cannot be changed by the user.
- Not Supported (N): Deprecated item

2 Reference

SI. No.	Title	Version
1	AUTOSAR_SWS_CryptoServiceManager.pdf	4.4.0
2	AUTOSAR_SWS_CryptoInterface.pdf	4.4.0
3	AUTOSAR_SWS_CryptoDriver.pdf	4.4.0
4	AUTOSAR_SWS_DefaultErrorTracer.pdf	4.4.0

3 AUTOSAR System

3.1 Crylf Module

This specification specifies the functionality, API and the configuration of the AUTOSAR Basic Software Module Crypto Interface (CRYIF).

The Crypto Interface module is located between the low level Crypto solutions (Crypto Driver and SW-based CDD) and the upper service layer (Crypto Service Manager). It represents the interface to the services of the Crypto Driver(s) for the upper service layer. A AUTOSAR Layered View can be found in *Figure 1*.

The Crypto Interface module provides a unique interface to manage different Crypto HW and SW solutions like HSM, SHE or SW-based CDD. Thus multiple underlying internal and external Crypto HW as well as SW solutions can be utilized by the Crypto Service Manager module based on a mapping scheme maintained by Crypto Interface.

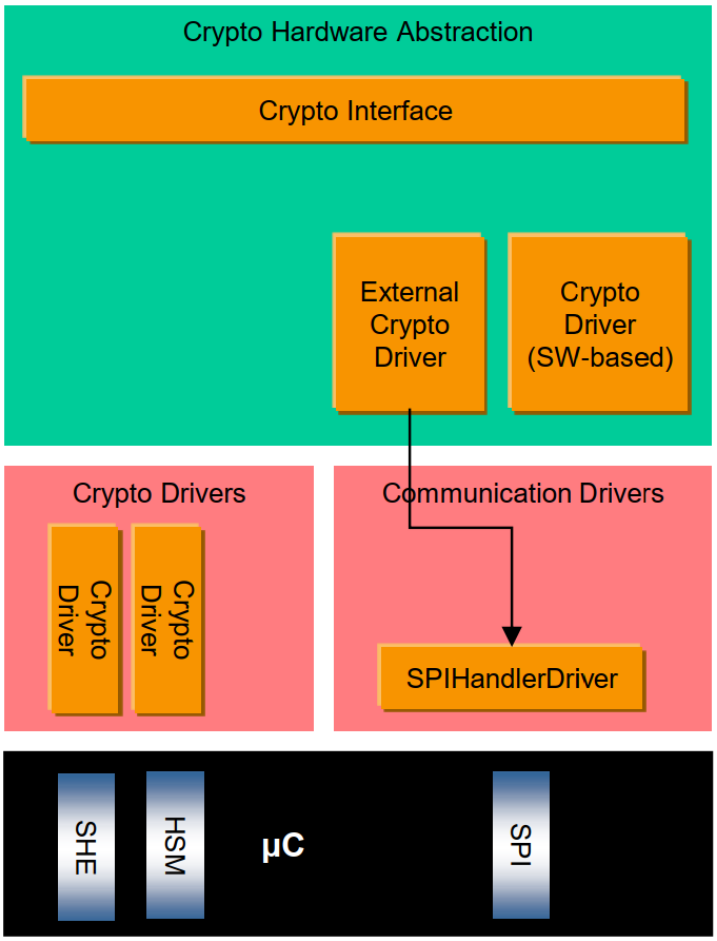

Figure 1

4 Product Release Notes

4.1 Overview

This chapter aims to provide the release information for the Hyundai Autoever CryIf module. Describes the limitations and specifics about the software product release version.

4.2 Scope of the Release

All information in this document is limited to the following Hyundai Autoever CryIf modules.

Module Name	AUTOSAR Version	Module Version
CryIf	4.4.0	1.0.5.0

Module version means Sw version of each module's BswModule Description (Bswmd) file.

4.3 Change Log

4.3.1 Version 1.0.0.0 (2021-01-15)

- **Version 1.0.0**
  - Initial Version

Cause	Initial Version
Operation Impact	N/A
Configuration Impact	N/A
Required measure of ASW	N/A

4.3.2 Version 1.0.1.0 (2021-03-20)

- **Version 1.0.1**
  - Fixed bug of CryIf_KeyElementCopyPartial function
  - Fixed bug of CryIf_KeyElementCopy function

Cause	CryIf_KeyElementCopy function has bug CryIf_KeyElementCopyPartial function has bug
Operation Impact	N/A
Configuration Impact	N/A
Required measure of ASW	N/A

4.3.3 Version 1.0.2.0 (2021-11-12)

- **Version 1.0.2**
  - Applying change of company name
  - Update CryIf_Version files to support R40 SWP compatible

Cause	Applying change of company name Update CryIf_Version files to support R40 SWP compatible
Operation Impact	N/A
Configuration Impact	N/A
Required measure of ASW	N/A

4.3.4 Version 1.0.2.1 (2022-07-01)

- **Change Request**
  - Change the Copyright comment in the code
  - DeliveryBoxHistory document template updates
  - Divide ‘delivery’ folder into ‘delivery/src’ and ‘delivery/inc’ folder

Cause	The new Copyright comment is needed to update in the code. The new DeliveryBoxHistory document template is needed to update. The ‘delivery’ folder should be divided into ‘delivery/src’ and ‘delivery/inc’ sub folders
Operation Impact	N/A
Configuration Impact	N/A
Required measure of ASW	N/A

4.3.5 Version 1.0.2.2 (2022-07-20)

- **Change Request**
  - Update EA, E-code and QT to make TM 100% coverage.
  - Update review template of SUD, SIT, SAD, SUT.

Cause	SAD<->SRS, SAD<->E-code and SRS<->UT traceability matrix are not covered fully. So, TM document does not reach 100% coverage. Review template of SUD, SIT, SAD, SUT is out of dated. It needs to update.
Operation Impact	N/A
Configuration Impact	N/A
Required measure of ASW	N/A

4.3.6 Version 1.0.3.0 (2022-08-23)

- **Change Request**
  - Fix UNECE violation coding security.

Cause	There are some violations against UNECE coding security rules. So, it needs to fix all of them.
Operation Impact	N/A
Configuration Impact	N/A
Required measure of ASW	N/A

4.3.7 Version 1.0.4.0 (2022-12-07)

- **Bug**
  - Correct the current implementation of CryIf_KeyCopy() API.

Cause	Defect Description : When NXP HSE Crypto Driver (Crypto_43) is integrated, CryIf_KeyCopy() return E_NOT_OK when the Target and Source are the same Crypto_43.

	Defect Causes : CryIf_KeyCopy() calls Crypto_KeyElementSet even though the target and source are the same driver. Defect Steps to reproduce : CryIf_KeyCopy() is invoked when source and target key are located NXP Crypto Driver (Crypto_43). Defect Correction : Correct the current implementation of CryIf_KeyCopy() API to follow exactly [SWS_CryIf_00120] (following ASR).
Operation Impact	N/A
Configuration Impact	N/A
Required measure of ASW	N/A

- **Bug**
  - Fix wrong generation of CryIf_Cfg.h when Vendor Api Infix is not set.

Cause	Defect Description : When Crypto Driver has <vi> and not has <ai>, CryIf_Cfg.h generate wrong Crypto header file name. Defect Causes : CryIf Generator does not consider <ai> absence. Defect Steps to reproduce : <vi> is set and <ai> is not set in the Crypto Driver Bswmd, wrong header file name is generated in CryIf_Cfg.h Defect Correction : Correct Generator to omit _<vi>_<ai> if CryptoDriver BSW module description does not set vendorApiInfix <vi> (following ASR Spec)
Operation Impact	N/A
Configuration Impact	N/A
Required measure of ASW	N/A

4.3.8 Version 1.0.5.0 (2023-03-03)

- **Improvement**
  - Improvement on including Header of Crypto Driver

Cause	Crypto Drivers can include each Crypto Driver's header through CryIf.h (CryIf_Cfg.h) so that header including needs to be moved to CryIf_Cfg.c
Operation Impact	N/A
Configuration Impact	N/A
Required measure of ASW	N/A

4.4 Module Release Notes

4.4.1 Limitations

- The Crypto Interface is specifically designed to operate with one or multiple underlying Crypto Drivers. Several Crypto Driver modules covering different HW processing units or cores are represented by just one generic interface as specified in the Crypto Driver specification. Any software based Crypto Driver shall be implemented as a CDD represented by the same interface above.

4.4.2 Deviations

- None

5 Configuration Guide

The CryIf setting of the AUTOSAR platform distributed by Hyundai Auto is a setting reflecting Hyundai Auto Policy's policy. Therefore, you should consult with Hyundai Auto.

5.1 CryIfGeneral Settings

Parameter Name	Value	Category
CryIfDevErrorDetect	User Defined	C
CryIfVersionInfoApi	User Defined	C

5.2 CryIfChannel Settings

Parameter Name	Value	Category
1)CryIfChannelId	User Defined	C
2)CryIfDriverObjectRef	CryptoDriverObject	C

1) CryIfChannelId value must be unique.
2) CryIfDriverObjectRef value must be unique and refer to CryptoDriverObject container.

5.3 CryIfKey Settings

Parameter Name	Value	Category
1) CryIfKeyId	User Defined	C
2)CryIfKeyRef	CryptoKey	C

1) CryIfKeyId value must be unique.
2) CryIfKeyRef value must be unique and refer to CryptoKey container.

5.4 Note

Before generation, the input needs all Bswmdt of Crypto Drivers.

All files Ecucd of Crypto Drivers must have different short name of ARpackage.

6 Application Programming Interface (API)

6.1 Type Definitions

6.1.1 CryIf_ConfigType

Type	Structure
Range	Implementation specific (The content of the configuration data structure is implementation specific).
Description	Configuration data structure of CryIf module.

6.2 Macro Constants

None

6.3 Functions

6.3.1 CryIf_Init

Function Name	CryIf_Init	
Syntax	void CryIf_Init(const CryIf_ConfigType* configPtr)	
Service ID [Hex]	0x00	
Sync/Async	Synchronous	
Reentrancy	Non-Reentrant	
Parameters (In)	configPtr	Pointer to a selected configuration structure.
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	None	
Description	Initializes the CRYIF module.	
Preconditions	None	
Configuration Dependency	None	
Available via	CryIf.h	

6.3.2 CryIf_GetVersionInfo

Function Name	CryIf_GetVersionInfo	
Syntax	void CryIf_GetVersionInfo(Std_VersionInfoType* versioninfo)	
Service ID [Hex]	0x01	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	versioninfo	Pointer to where to store the version information of this module.
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	None	
Description	Returns the version information of this module.	
Preconditions	None	

Configuration Dependency	CryIfVersionInfoApi
Available via	CryIf.h

6.3.3 CryIf_ProcessJob

Function Name	CryIf_ProcessJob	
Syntax	Std_ReturnType CryIf_ProcessJob(uint32 channelId, Crypto_JobType* job)	
Service ID [Hex]	0x03	
Sync/Async	Sync or Async, depends on the configuration	
Reentrancy	Reentrant	
Parameters (In)	channelId	Holds the identifier of the crypto channel.
Parameters (Inout)	job	Pointer to the configuration of the job. Contains structures with user and primitive relevant information.
Parameters (Out)	None	
Return Value	Std_ReturnType	E_OK: Request successful E_NOT_OK: Request failed CRYPTO_E_BUSY: Request failed, Crypro Driver Object is busy CRYPTO_E_KEY_NOT_VALID: Request failed, the key is not valid CRYPTO_E_KEY_SIZE_MISMATCH: Request failed, a key element has the wrong size CRYPTO_E_QUEUE_FULL: Request failed, the queue is full CRYPTO_E_KEY_READ_FAIL: The service request failed, because key element extraction is not allowed CRYPTO_E_KEY_WRITE_FAIL: The service request failed because the writing access failed CRYPTO_E_KEY_NOT_AVAILABLE: The service request failed because the key is not available CRYPTO_E_SMALL_BUFFER: The provided buffer is too small to store the result CRYPTO_E_JOB_CANCELED: The service request failed because the synchronous Job has been canceled CRYPTO_E_KEY_EMPTY: Request failed because of uninitialized source key element
Description	This interface dispatches the received jobs to the configured crypto driver object.	
Preconditions	None	
Configuration Dependency	None	
Available via	CryIf.h	

6.3.4 CryIf_CancelJob

Function Name	CryIf_CancelJob
Syntax	Std_ReturnType CryIf_CancelJob(uint32 channelId, Crypto_JobType* job)

Service ID [Hex]	0x0e	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	channelId	Holds the identifier of the crypto channel.
Parameters (Inout)	job	Pointer to the configuration of the job. Contains structures with user and primitive relevant information.
Parameters (Out)	None	
Return Value	Std_ReturnType	E_OK: Request successful, job has been removed E_NOT_OK: Request failed, job couldn't be removed
Description	This interface dispatches the job cancellation function to the configured crypto driver object.	
Preconditions	None	
Configuration Dependency	None	
Available via	CryIf.h	

6.3.5 CryIf_KeyElementSet

Function Name	CryIf_KeyElementSet	
Syntax	Std_ReturnType CryIf_KeyElementSet(uint32 cryIfKeyId, uint32 keyElementId, const uint8* keyPtr, uint32 keyLength)	
Service ID [Hex]	0x04	
Sync/Async	Synchronous	
Reentrancy	Non-Reentrant	
Parameters (In)	cryIfKeyId	Holds the identifier of the key whose key element shall be set.
	keyElementId	Holds the identifier of the key element which shall be set.
	keyPtr	Holds the pointer to the key data which shall be set as key element.
	keyLength	Contains the length of the key element in bytes.
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	Std_ReturnType	E_OK: Request successful E_NOT_OK: Request failed CRYPTO_E_BUSY: Request failed, Crypto Driver Object is busy CRYPTO_E_KEY_WRITE_FAIL: Request failed because write access was denied CRYPTO_E_KEY_NOT_AVAILABLE: Request failed because the key is not available CRYPTO_E_KEY_SIZE_MISMATCH: Request failed, key element size does not match size of provided data
Description	This function shall dispatch the set key element function to the configured crypto driver object.	
Preconditions	None	
Configuration Dependency	None	
Available via	CryIf.h	

6.3.6 CryIf_KeySetValid

Function Name	CryIf_KeySetValid	
Syntax	Std_ReturnType CryIf_KeySetValid(uint32 cryIfKeyId)	
Service ID [Hex]	0x05	
Sync/Async	Synchronous	
Reentrancy	Non-Reentrant	
Parameters (In)	cryIfKeyId	Holds the identifier of the key whose key elements shall be set to valid.
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	Std_ReturnType	E_OK: Request successful E_NOT_OK: Request failed CRYPTO_E_BUSY: Request failed, Crypro Driver Object is busy
Description	This function shall dispatch the set key valid function to the configured crypto driver object.	
Preconditions	None	
Configuration Dependency	None	
Available via	CryIf.h	

6.3.7 CryIf_KeyElementGet

Function Name	CryIf_KeyElementGet	
Syntax	Std_ReturnType CryIf_KeyElementGet(uint32 cryIfKeyId, uint32 keyElementId, uint8* resultPtr, uint32* resultLengthPtr)	
Service ID [Hex]	0x06	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	cryIfKeyId	Holds the identifier of the key whose key element shall be returned.
	keyElementId	Holds the identifier of the key element which shall be returned.
Parameters (Inout)	resultLengthPtr	Holds a pointer to a memory location in which the length information is stored. On calling this function this parameter shall contain the size of the buffer provided by resultPtr. If the key element is configured to allow partial access, this parameter contains the amount of data which should be read from the key element. The size may not be equal to the size of the provided buffer anymore. When the request has finished, the amount of data that has been stored shall be stored.
Parameters (Out)	resultPtr	Holds the pointer of the buffer for the returned key element.
Return Value	Std_ReturnType	E_OK: Request successful E_NOT_OK: Request failed CRYPTO_E_BUSY: Request failed, Crypto Driver Object is busy

		CRYPTO_E_KEY_NOT_AVAILABLE: Request failed, the requested key element is not available CRYPTO_E_KEY_READ_FAIL: Request failed because read access was denied CRYPTO_E_SMALL_BUFFER: The provided buffer is too small to store the result CRYPTO_E_KEY_EMPTY: Request failed because of uninitialized source key element
Description	This function shall dispatch the get key element function to the configured crypto driver object.	
Preconditions	None	
Configuration Dependency	None	
Available via	CryIf.h	

6.3.8 CryIf_KeyElementCopy

Function Name	CryIf_KeyElementCopy	
Syntax	Std_ReturnType CryIf_KeyElementCopy(uint32 cryIfKeyId, uint32 keyElementId, uint32 targetCryIfKeyId, uint32 targetKeyElementId)	
Service ID [Hex]	0x0f	
Sync/Async	Synchronous	
Reentrancy	Reentrant, but not for the same cryIfKeyId	
Parameters (In)	cryIfKeyId	Holds the identifier of the key whose key element shall be the source element.
	keyElementId	Holds the identifier of the key element which shall be the source for the copy operation.
	targetCryIfKeyId	Holds the identifier of the key whose key element shall be the destination element.
	targetKeyElementId	Holds the identifier of the key element which shall be the destination for the copy operation.
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	Std_ReturnType	E_OK: Request successful E_NOT_OK: Request failed CRYPTO_E_BUSY: Request failed, Crypto Driver Object is busy CRYPTO_E_KEY_NOT_AVAILABLE: Request failed, the requested key element is not available CRYPTO_E_KEY_READ_FAIL: Request failed, not allowed to extract key element CRYPTO_E_KEY_WRITE_FAIL: Request failed, not allowed to write key element CRYPTO_E_KEY_SIZE_MISMATCH: Request failed, key element sizes are not compatible CRYPTO_E_KEY_EMPTY: Request failed because of uninitialized source key element
Description	This function shall copy a key elements from one key to a target key.	
Preconditions	None	
Configuration	None	

Dependency	
Available via	CryIf.h

6.3.9 CryIf_KeyElementCopyPartial

Function Name	CryIf_KeyElementCopyPartial	
Syntax	Std_ReturnType CryIf_KeyElementCopyPartial(uint32 cryIfKeyId, uint32 keyElementId, uint32 keyElementSourceOffset, uint32 keyElementTargetOffset, uint32 keyElementCopyLength, uint32 targetCryIfKeyId, uint32 targetKeyElementId)	
Service ID [Hex]	0x12	
Sync/Async	Synchronous	
Reentrancy	Reentrant, but not for the same cryIfKeyId	
Parameters (In)	cryIfKeyId	Holds the identifier of the key whose key element shall be the source element.
	keyElementId	Holds the identifier of the key element which shall be the source for the copy operation.
	keyElementSourceOffset	This is the offset of the source key element indicating the start index of the copy operation.
	keyElementTargetOffset	This is the offset of the target key element indicating the start index of the copy operation.
	keyElementCopyLength	Specifies the number of bytes that shall be copied.
	targetCryIfKeyId	Holds the identifier of the key whose key element shall be the destination element.
	targetKeyElementId	Holds the identifier of the key element which shall be the destination for the copy operation.
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	Std_ReturnType	E_OK: Request successful E_NOT_OK: Request failed E_BUSY: Request failed, Crypto Driver Object is busy CRYPTO_E_KEY_NOT_AVAILABLE: Request failed, the requested key element is not available CRYPTO_E_KEY_READ_FAIL: Request failed, not allowed to extract key element CRYPTO_E_KEY_WRITE_FAIL: Request failed, not allowed to write key element CRYPTO_E_KEY_SIZE_MISMATCH: Request failed, key element sizes are not compatible CRYPTO_E_KEY_EMPTY: Request failed because of uninitialized source key element
Description	Copies a key element to another key element. The keyElementOffsets and keyElementCopyLength allows to copy just parts of the source key element into the destination key element.	
Preconditions	None	
Configuration Dependency	None	
Available via	CryIf.h	

6.3.10 CryIf_KeyCopy

Function Name	CryIf_KeyCopy	
Syntax	Std_ReturnType CryIf_KeyCopy(uint32 cryIfKeyId, uint32 targetCryIfKeyId)	
Service ID [Hex]	0x10	
Sync/Async	Synchronous	
Reentrancy	Reentrant, but not for the same cryIfKeyId	
Parameters (In)	cryIfKeyId	Pointer to a selected configuration structure.
	targetCryIfKeyId	Holds the identifier of the key whose key element shall be the destination element.
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	Std_ReturnType	E_OK: Request successful E_NOT_OK: Request failed E_BUSY: Request failed, Crypto Driver Object is busy CRYPTO_E_KEY_NOT_AVAILABLE: Request failed, the requested key element is not available CRYPTO_E_KEY_READ_FAIL: Request failed, not allowed to extract key element CRYPTO_E_KEY_WRITE_FAIL: Request failed, not allowed to write key element CRYPTO_E_KEY_SIZE_MISMATCH: Request failed, key element sizes are not compatible CRYPTO_E_KEY_EMPTY: Request failed because of uninitialized source key element
Description	This function shall copy all key elements from the source key to a target key.	
Preconditions	None	
Configuration Dependency	None	
Available via	CryIf.h	

6.3.11 CryIf_RandomSeed

Function Name	CryIf_RandomSeed	
Syntax	Std_ReturnType CryIf_RandomSeed(uint32 cryIfKeyId, const uint8* seedPtr, uint32 seedLength)	
Service ID [Hex]	0x07	
Sync/Async	Sync or Async, depends on the configuration	
Reentrancy	Reentrant	
Parameters (In)	Reentrant	Holds the identifier of the key for which a new seed shall be generated.
	seedPtr	Holds a pointer to the memory location which contains the data to feed the seed.
	seedLength	Contains the length of the seed in bytes.
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	Std_ReturnType	E_OK: Request successful

		E_NOT_OK: Request failed
Description	This function shall dispatch the random seed function to the configured crypto driver object.	
Preconditions	None	
Configuration Dependency	None	
Available via	CryIf.h	

6.3.12 CryIf_KeyGenerate

Function Name	CryIf_KeyGenerate	
Syntax	Std_ReturnType CryIf_KeyGenerate(uint32 cryIfKeyId)	
Service ID [Hex]	0x08	
Sync/Async	Sync or Async, depends on the configuration	
Reentrancy	Reentrant	
Parameters (In)	cryIfKeyId	Holds the identifier of the key which is to be updated with the generated value.
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	Std_ReturnType	E_OK: Request successful E_NOT_OK: Request failed E_BUSY: Request failed, Crypto Driver Object is busy CRYPTO_E_KEY_EMPTY: Request failed because of uninitialized source key element
Description	This function shall dispatch the key generate function to the configured crypto driver object.	
Preconditions	None	
Configuration Dependency	None	
Available via	CryIf.h	

6.3.13 CryIf_KeyDerive

Function Name	CryIf_KeyDerive	
Syntax	Std_ReturnType CryIf_KeyDerive(uint32 cryIfKeyId, uint32 targetCryIfKeyId)	
Service ID [Hex]	0x09	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	cryIfKeyId	Holds the identifier of the key which is used for key derivation.
	targetCryIfKeyId	Holds the identifier of the key which is used to store the derived key.
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	Std_ReturnType	E_OK: Request successful E_NOT_OK: Request failed CRYPTO_E_KEY_EMPTY: Request failed because of uninitialized source key element

Description	This function shall dispatch the key derive function to the configured crypto driver object.
Preconditions	None
Configuration Dependency	None
Available via	CryIf.h

6.3.14 CryIf_KeyExchangeCalcPubVal

Function Name	CryIf_KeyExchangeCalcPubVal	
Syntax	Std_ReturnType CryIf_KeyExchangeCalcPubVal(uint32 cryIfKeyId, uint8* publicValuePtr, uint32* publicValueLengthPtr)	
Service ID [Hex]	0x0a	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	cryIfKeyId	Holds the identifier of the key which shall be used for the key exchange protocol.
Parameters (Inout)	publicValueLengthPtr	Holds a pointer to the memory location in which the public value length information is stored. On calling this function, this parameter shall contain the size of the buffer provided by publicValuePtr. When the request has finished, the actual length of the returned value shall be stored.
Parameters (Out)	publicValuePtr	Contains the pointer to the data where the public value shall be stored.
Return Value	Std_ReturnType	E_OK: Request successful E_NOT_OK: Request failed E_BUSY: Request failed, Crypto Driver Object is busy CRYPTO_E_SMALL_BUFFER: The provided buffer is too small to store the result CRYPTO_E_KEY_EMPTY: Request failed because of uninitialized source key element
Description	This function shall dispatch the key exchange public value calculation function to the configured crypto driver object.	
Preconditions	None	
Configuration Dependency	None	
Available via	CryIf.h	

6.3.15 CryIf_KeyExchangeCalcSecret

Function Name	CryIf_KeyExchangeCalcSecret
Syntax	Std_ReturnType CryIf_KeyExchangeCalcSecret(uint32 cryIfKeyId, const uint8* partnerPublicValuePtr, uint32 partnerPublicValueLength)
Service ID [Hex]	0x0b

Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	cryIfKeyId	Holds the identifier of the key which shall be used for the key exchange protocol.
	partnerPublicValuePtr	Holds the pointer to the memory location which contains the partner's public value.
	partnerPublicValueLength	Contains the length of the partner's public value in bytes.
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	Std_ReturnType	E_OK: Request successful E_NOT_OK: Request failed E_BUSY: Request failed, Crypto Driver Object is busy CRYPTO_E_SMALL_BUFFER: The provided buffer is too small to store the result CRYPTO_E_KEY_EMPTY: Request failed because of uninitialized source key element
Description	This function shall dispatch the key exchange common shared secret calculation function to the configured crypto driver object.	
Preconditions	None	
Configuration Dependency	None	
Available via	CryIf.h	

6.3.16 CryIf_CallbackNotification

Function Name	CryIf_CallbackNotification	
Syntax	void CryIf_CallbackNotification(Crypto_JobType* job, Csm_ResultType result)	
Service ID [Hex]	0x0d	
Sync/Async	Synchronous	
Reentrancy	Non-Reentrant	
Parameters (In)	job	Points to the completed job's information structure. It contains a callbackID to identify which job is finished.
	result	Contains the result of the cryptographic operation.
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	None	
Description	Notifies the CRYIF about the completion of the request with the result of the cryptographic operation.	
Preconditions	None	
Configuration Dependency	None	
Available via	CryIf.h	

6.3.17 Note

Two functions CryIf_CertificateParse and CryIf_CertificateVerify are not supported by CryIf. All of certificate services shall be supported by KeyM.

7 Generator

7.1 Generator Option

Options	Description
-G,--Generation	Symbolic parameters to be used for fore generation (skip validation).
-I,--Input <I>	ECU description file path of the module for which generation tool need to run.
-L,--Log	Symbolic parameters to be used for generation error log.
-M,--Module <M>	Specify module name and version to be generated code for.
-O,--Output <O>	Project-relative path to location where the generated code is to be placed.
-T,--Top_path <T>	Symbolic parameters to be used for set path of module.
-V,--Validate	Symbolic parameters to be used for invoking validation checks.

7.2 Generator Error Message

ERR112001 The value configured for parameter MODULE-ID in container BSW-MODULE-DESCRIPTION in provided MDT file is not correct. Module ID of CryIf must be 112.

This error message is displayed if the value of ModuleId in file BSWMDT is not equal with the ModuleId of CryIf.

ERR112002 The value configured for parameter VENDOR-ID in container BSW-IMPLEMENTATION in provided MDT file is not correct. Vendor ID of CryIf must be 76.

This error message is displayed if the value of VendorId in file BSWMDT is not equal with the VendorId of CryIf.

ERR112003 The parameter <Parameter Name> in the container <Container Name> should be configured.

This error message is displayed if any of the mandatory configuration parameters mentioned below is not configured in ECU Configuration Description File.

Parameter name	Container name
AR-RELEASE-VERSION	BSW-IMPLEMENTATION
SW-VERSION	BSW-IMPLEMENTATION
VENDOR-ID	BSW-IMPLEMENTATION
MODULE-ID	BSW-MODULE-DESCRIPTION

ERR112004 The value configured parameter <Parameter Name> in the container <Container Name> is incorrect. It should follow the example.

This error message is displayed if the parameters 'Parameter Name' is not configured as per the pattern.

Parameter name	Container name	Pattern
SW-VERSION	BSW-IMPLEMENTATION	[0-9]+.[0-9]+.[0-9]+

ERR112005 AUTOSAR RELEASE VERSION <Version> configured for the parameter <AR-RELEASE-VERSION> in provided MDT file is not correct. AUTOSAR RELEASE VERSION should be 4.4.0.

This error message is displayed if the value of the element AR-RELEASE-VERSION present in file BSWMDT is configured other than 4.4.0.

ERR112006 The parameter <Parameter Name> in the container <Container Name> must unique.

This error message is displayed if the value of the parameters 'Parameter Name' mentioned below is not unique.

Parameter name	Container name
CryIfChannelId	CryIfChannel
CryIfDriverObjectRef	CryIfChannel
CryIfKeyId	CryIfKey
CryIfKeyRef	CryIfKey

8 Appendix