Document Title	SWS_CryptoDriver: Complete Change Documentation 4.3.0 - 4.3.1
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	695

Document Status	Final
Part of AUTOSAR Standard	Classic Platform
Part of Standard Release	4.3.1

Table of Contents

1 SWS_CryptoDriver

1.1 Specification Item ECUC_Crypto_00002

Trace References:

none

Content:

Container Name	CryptoGeneralCryptoGeneral		
Description	Container for common configuration options		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

Included parameters:

Included Parameters	
Parameter Name	SWS Item ID
CryptoDevErrorDetect	ECUC_Crypto_00006
CryptoInstanceId	ECUC_Crypto_00040
CryptoMainFunctionPeriod	ECUC_Crypto_00038
CryptoVersionInfoApi	ECUC_Crypto_00007

Included containers:

No Included Containers

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77845: [diverse] Configuration of instance ID for instantiated modules

  Problem description:

  Some modules that can exist multiple times in an AUTOSAR BSW stack have configurable instance IDs that are used to e.g. call DET. Examples are the bus drivers. Others, like the CDD, Crypto driver, or DIO driver, lack such a configuration parameter.
  –Last change on issue 77845 comment 2–

Agreed solution:

TPS EcuConfigurationSpecification (CDD):

Add container CddGeneral with one parameter CddInstanceId to Cdd EcucModuleDef
Description: Specifies the InstanceId of this module instance. If only one instance is present it shall have the Id 0.
Multiplicity: 1
Type: EcucIntegerParamDef
Range: 0 .. 255
Default value: -
Post-Build Variant Value: false
Value Configuration Class: Pre-compile time - All Variants
==
Crypto:

Add new pre-compile integer parameter "CryptoInstanceId" (range 0..255) to the container CryptoGeneral,
Description: "Instance ID of the crypto driver. This ID is used to discern several crypto drivers in case more than one driver is used in the same ECU."
==
Eep:

Change ECUC_Eep_00189 Description from : "Represents the Index of the driver, used by EA" to
"Specifies the InstanceId of this module instance. If only one instance is present it shall have the Id 0."
==
Fls:

Add REQ:
SWS_Fls_xxx: If more than one instance of the flash driver is used in an ECU, the individual instances have to be given a unique instance ID. This instance ID shall be configured as the parameter FlsDriverIndex. If only one instance of the flash driver is used in an ECU, this instance shall have the parameter FlsDriverIndex configured as 0.
==
Wdg:

Change ECUC_Wdg_00117 Description from : "Represents the watchdog driver's ID so that it can be referenced by the watchdog interface." to
"Specifies the InstanceId of this module instance. If only one instance is present it

shall have the Id 0."
===
Xfrm:

Add into the container XfrmGeneral a new parameter XfrmInstanceId to Xfrm EcucModuleDef
Description: Specifies the InstanceId of this module instance. If only one instance is present it shall have the Id 0.
Multiplicity: 1
Type: EcucIntegerParamDef
Range: 0 .. 255
Default value: -
Post-Build Variant Value: false
Value Configuration Class: Pre-compile time - All Variants
–Last change on issue 77845 comment 32–

BW-C-Level:

Application	Specification	Bus
1	3	1

1.2 Specification Item ECUC_Crypto_00014

Trace References:

none

Content:

Container Name	CryptoKeyElementCryptoKeyElement
Description	Configuration of a CryptoKeyElement
Configuration Parameters	

Included parameters:

Included Parameters	
Parameter Name	SWS Item ID
CryptoKeyElementAllowPartialAccess	ECUC_Crypto_00025
CryptoKeyElementFormat	ECUC_Crypto_00041
CryptoKeyElementId	ECUC_Crypto_00021
CryptoKeyElementInitValue	ECUC_Crypto_00023
CryptoKeyElementPersist	ECUC_Crypto_00026
CryptoKeyElementReadAccess	ECUC_Crypto_00024

Included Parameters	
Parameter Name	SWS Item ID
CryptoKeyElementSize	ECUC_Crypto_00022
CryptoKeyElementWriteAccess	ECUC_Crypto_00027
CryptoKeyElementVirtualTargetRef	ECUC_Crypto_00028

Included containers:

No Included Containers

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77661: Definition for asymmetric key formats

  Problem description:

  Name: Armin Happel

  Description/Motivation:
  Currently, the AUTOSAR crypto stack specifies to provide asymmetric key material in PKCS# 8 format only [see SWS_CSM_00951]. However, the standard is not precise enough and defines only the usage of private key material. Optionally, public key material can be provided in addition. This provides the lack of definition in the AUTOSAR stack, that public keys cannot be provided for certain algorithms, such as signature verification.
  This RFC extends the current definition so that also public key material can be provided to the crypto stack.

  Agreed solution:

  See attachment: https://bugzilla.autosar.org/attachment.cgi?id=4617
  –Last change on issue 77661 comment 29–

  BW-C-Level:

Application	Specification	Bus
1	4	1

1.3 Specification Item ECUC_Crypto_00018

Trace References:

none

Content:

Name	CryptoPrimitiveRefCryptoDriverObject.CryptoPrimitiveRefin container CryptoDriverObject		
Description	Refers to primitive in the CRYPTO. The CryptoPrimitive is a pre-configured container of the crypto service that shall be used.		
Multiplicity	1 1..*		
Type	Reference to [CryptoPrimitive]		
Post-Build Variant Multiplicity	false		
Post-Build Variant Value	false		
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77304: [CRYPTO] Specification of configurations of CryptoPrimitive (ECUC_Crypto_00033) is not sensible

  Problem description:

  Container CryptoPrimitive (ECUC_Crypto_00033) contains
  - 1..1 parameters CryptoPrimitiveService
  - 1..* parameters CryptoPrimitiveAlgorithmFamiliy
  - 1..* parameters CryptoPrimitiveAlgorithmMode
  - 1..* parameters CryptoPrimitiveAlgorithmSecondaryFamiliy
  This results in incoherent sets of algorithm families and modes per CryptoPrimitive and an improper service very probably.
  How shall be determined (by the CryIf) which modes are valid for which familiy, especially if there are several families each supporting another specific mode of the same group of modes.
  Why is CryptoPrimitiveService necessary?

  E.g.
  CryptoPrimitive =
  - CryptoPrimitiveService
  - MAC_GENERATE
  - CryptoPrimitiveAlgorithmFamiliy
  + CRYPTO_ALGOFAM_3DES

+ CRYPTO_ALGOFAM_AES
+ CRYPTO_ALGOFAM_SHA1
- CryptoPrimitiveAlgorithmMode
+ CRYPTO_ALGOMODE_ECB (<– only valid for CRYPTO_ALGOFAM_3DES)
+ CRYPTO_ALGOMODE_CBC (<– only valid for CRYPTO_ALGOFAM_AES)
+ CRYPTO_ALGOMODE_XTS
+ CRYPTO_ALGOMODE_RSASSA_PSS
- CryptoPrimitiveAlgorithmSecondaryFamiliy
+ CRYPTO_ALGOFAM_NOT_SET

On the other hand this is not a "Configuration of a [clear] CryptoPrimitive" as the description of ECUC_Crypto_00033 specifies, but a set of primitives provided by a CryptoDriverObject.

Proposal:
CryptoPrimitive contains
- 1..1 parameters CryptoPrimitiveAlgorithmFamiliy
- 1..1 parameters CryptoPrimitiveAlgorithmMode
- 1..1 parameters CryptoPrimitiveAlgorithmSecondaryFamiliy
CryptoDriverObject contains
- 1..* parameters CryptoPrimitiveRef (or 1..* containers CryptoPrimitiveRefContainer with single parameter 1..1 CryptoPrimitiveRef)

E.g.
CryptoPrimitive_0 =
- CryptoPrimitiveAlgorithmFamiliy
+ CRYPTO_ALGOFAM_3DES
- CryptoPrimitiveAlgorithmMode
+ CRYPTO_ALGOMODE_ECB
- CryptoPrimitiveAlgorithmSecondaryFamiliy
+ CRYPTO_ALGOFAM_NOT_SET
CryptoPrimitive_1 =
- CryptoPrimitiveAlgorithmFamiliy
+ CRYPTO_ALGOFAM_AES
- CryptoPrimitiveAlgorithmMode
+ CRYPTO_ALGOMODE_CBC
- CryptoPrimitiveAlgorithmSecondaryFamiliy
+ CRYPTO_ALGOFAM_NOT_SET
CryptoPrimitive_2 =
- CryptoPrimitiveAlgorithmFamiliy
+ CRYPTO_ALGOFAM_SHA1

- CryptoPrimitiveAlgorithmMode
+ CRYPTO_ALGOMODE_NOT_SET
- CryptoPrimitiveAlgorithmSecondaryFamiliy
+ CRYPTO_ALGOFAM_NOT_SET

CryptoDriverObject
- CryptoPrimitiveRef
+ CryptoPrimitive_0
+ CryptoPrimitive_1
+ CryptoPrimitive_2

Agreed solution:

Extracted from Problem Description:

ECUC_Crypto_00035: Adapt multiplicity to 1..1
ECUC_Crypto_00036: Adapt multiplicity to 1..1
ECUC_Crypto_00037: Adapt multiplicity to 1..1
ECUC_Crypto_00018: Adapt multiplicity to 1..*
–Last change on issue 77304 comment 3–

BW-C-Level:

Application	Specification	Bus
1	4	1

1.4 Specification Item ECUC_Crypto_00028

Trace References:

none

Content:

Name	CryptoKeyElementVirtualTargetRefCryptoKeyElement.CryptoKeyElementVirtualTargetRefin container CryptoKeyElement		
Description	Refers to a key element which will contain the actual data. If the Reference is configured, the key element will be a virtual key element.		
Multiplicity	1..* 0..1		
Type	Reference to [CryptoKeyElement]		
Post-Build Variant Value	true		
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	

Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77579: [CRYPTO] Wrong multiplicity for CryptoKeyElementVirtualTargetRef in CryptoKeyElement

  Problem description:

  Hi,

  ECUC_Crypto_00028 states that the reference to a key element which will contain the actual data (CryptoKeyElementVirtualTargetRef) has multiplicity 1..*. But it should by 0..1.

  As soon as there is an entry for CryptoKeyElementVirtualTargetRef, the key element will be a virtual key element. Not all key elements are virtual. That's why it should have lower multiplicity 0. And the upper multiplicity should be 1, because a reference to the key element with the actual data is a 1:1 relation.

  Best Regards,
  Petra Elas-Welter

  Agreed solution:

  ECUC_Crypto_00028: Adapt multiplicity to 0..1
  –Last change on issue 77579 comment 3–

  BW-C-Level:

Application	Specification	Bus
1	1	1

1.5 Specification Item ECUC_Crypto_00033

Trace References:

none

Content:

Container Name	CryptoPrimitiveCryptoPrimitive		
Description	Configuration of a CryptoPrimitive		
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

Included parameters:

Included Parameters	
Parameter Name	SWS Item ID
CryptoPrimitiveAlgorithmFamiliy Family	ECUC_Crypto_00035
CryptoPrimitiveAlgorithmMode	ECUC_Crypto_00036
CryptoPrimitiveAlgorithmSecondaryFamiliy Family	ECUC_Crypto_00037
CryptoPrimitiveService	ECUC_Crypto_00034

Included containers:

No Included Containers

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77711: [CRYPTO] Csm<Service>AlgorithmFamiliy

  Problem description:

  The name of all configuration parameters CsmHash|MacGenerate|MacVerify|...AlgorithmFamiliy is not correctly written.
  There is an "i" before the "y" in "Family".

  RfC 76783 mentioned this for CsmMacGenerateAlgorithmFamiliy only.

  Agreed solution:

  Change Csm<Service>AlgorithmFamiliy to Csm<Service>AlgorithmFamily in the following ECUCs:
  ECUC_Csm_00038
  ECUC_Csm_00188
  ECUC_Csm_00051
  ECUC_Csm_00182
  ECUC_Csm_00066
  ECUC_Csm_00074
  ECUC_Csm_00082

ECUC_Csm_00089
ECUC_Csm_00096
ECUC_Csm_00105

SWS_CryptoDriver:
Change Familiy to Family:
ECUC_Crypto_00035
ECUC_Crypto_00037
–Last change on issue 77711 comment 8–

BW-C-Level:

Application	Specification	Bus
1	3	1

1.6 Specification Item ECUC_Crypto_00035

Trace References:

none

Content:

Name	CryptoPrimitiveAlgorithmFamiliyFamilyCryptoPrimitive.CryptoPrimitiveAlgorithmFamiliyin container Family
Parent Container	CryptoPrimitive
Description	Determines the algorithm family used for the crypto service
Multiplicity	1..* 1
Type	EcucEnumerationParamDef

Range	CRYPTO_ALGOFAM_3DES	0x13	CryptoPrimitive.CryptoPrimitiveAlgorithmFamiliyFamily.CRYPTO_ALGOFAM_3DES
	CRYPTO_ALGOFAM_AES	0x14	CryptoPrimitive.CryptoPrimitiveAlgorithmFamiliyFamily.CRYPTO_ALGOFAM_AES
	CRYPTO_ALGOFAM_BLAKE_1_256	0x0F	CryptoPrimitive.CryptoPrimitiveAlgorithmFamiliyFamily.CRYPTO_ALGOFAM_BLAKE_1_256
	CRYPTO_ALGOFAM_BLAKE_1_512	0x10	CryptoPrimitive.CryptoPrimitiveAlgorithmFamiliyFamily.CRYPTO_ALGOFAM_BLAKE_1_512
	CRYPTO_ALGOFAM_BLAKE_2s_256	0x11	CryptoPrimitive.CryptoPrimitiveAlgorithmFamiliyFamily.CRYPTO_ALGOFAM_BLAKE_2s_256
	CRYPTO_ALGOFAM_BLAKE_2s_512	0x12	CryptoPrimitive.CryptoPrimitiveAlgorithmFamiliyFamily.CRYPTO_ALGOFAM_BLAKE_2s_512
	CRYPTO_ALGOFAM_BRAINPOOL	0x18	CryptoPrimitive.CryptoPrimitiveAlgorithmFamiliyFamily.CRYPTO_ALGOFAM_BRAINPOOL
	CRYPTO_ALGOFAM_CHACHA	0x15	CryptoPrimitive.CryptoPrimitiveAlgorithmFamiliyFamily.CRYPTO_ALGOFAM_CHACHA
	CRYPTO_ALGOFAM_CUSTOM	0xFF	CryptoPrimitive.CryptoPrimitiveAlgorithmFamiliyFamily.CRYPTO_ALGOFAM_CUSTOM
	CRYPTO_ALGOFAM_ECCNIST	0x19	CryptoPrimitive.CryptoPrimitiveAlgorithmFamiliyFamily.CRYPTO_ALGOFAM_ECCNIST
	CRYPTO_ALGOFAM_ECIES	0x1D	CryptoPrimitive.CryptoPrimitiveAlgorithmFamiliyFamily.CRYPTO_ALGOFAM_ECIES
	CRYPTO_ALGOFAM_ED25519	0x17	CryptoPrimitive.CryptoPrimitiveAlgorithmFamiliyFamily.CRYPTO_ALGOFAM_ED25519
	CRYPTO_ALGOFAM_NOT_SET	0x00	CryptoPrimitive.CryptoPrimitiveAlgorithmFamiliyFamily.CRYPTO_ALGOFAM_NOT_SET
	CRYPTO_ALGOFAM_RIPEMD160	0x0E	CryptoPrimitive.CryptoPrimitiveAlgorithmFamiliyFamily.CRYPTO_ALGOFAM_RIPEMD160
	CRYPTO_ALGOFAM_RNG	0x1B	CryptoPrimitive.CryptoPrimitiveAlgorithmFamiliyFamily.CRYPTO_ALGOFAM_RNG
	CRYPTO_ALGOFAM_RSA	0x16	Crypto

Post-Build Variant Value	false		
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77304: [CRYPTO] Specification of configurations of CryptoPrimitive (ECUC_Crypto_00033) is not sensible

  Problem description:

  Container CryptoPrimitive (ECUC_Crypto_00033) contains
  - 1..1 parameters CryptoPrimitiveService
  - 1..* parameters CryptoPrimitiveAlgorithmFamiliy
  - 1..* parameters CryptoPrimitiveAlgorithmMode
  - 1..* parameters CryptoPrimitiveAlgorithmSecondaryFamiliy
  This results in incoherent sets of algorithm families and modes per CryptoPrimitive and an improper service very probably.
  How shall be determined (by the CryIf) which modes are valid for which familiy, especially if there are several families each supporting another specific mode of the same group of modes.
  Why is CryptoPrimitiveService necessary?

  E.g.
  CryptoPrimitive =
  - CryptoPrimitiveService
  - MAC_GENERATE
  - CryptoPrimitiveAlgorithmFamiliy
  + CRYPTO_ALGOFAM_3DES
  + CRYPTO_ALGOFAM_AES
  + CRYPTO_ALGOFAM_SHA1
  - CryptoPrimitiveAlgorithmMode
  + CRYPTO_ALGOMODE_ECB (<– only valid for CRYPTO_ALGOFAM_3DES)
  + CRYPTO_ALGOMODE_CBC (<– only valid for CRYPTO_ALGOFAM_AES)
  + CRYPTO_ALGOMODE_XTS
  + CRYPTO_ALGOMODE_RSASSA_PSS
  - CryptoPrimitiveAlgorithmSecondaryFamiliy
  + CRYPTO_ALGOFAM_NOT_SET

On the other hand this is not a "Configuration of a [clear] CryptoPrimitive" as the description of ECUC_Crypto_00033 specifies, but a set of primitives provided by a CryptoDriverObject.

Proposal:
CryptoPrimitive contains
- 1..1 parameters CryptoPrimitiveAlgorithmFamiliy
- 1..1 parameters CryptoPrimitiveAlgorithmMode
- 1..1 parameters CryptoPrimitiveAlgorithmSecondaryFamiliy
CryptoDriverObject contains
- 1..* parameters CryptoPrimitiveRef (or 1..* containers CryptoPrimitiveRefContainer with single parameter 1..1 CryptoPrimitiveRef)

E.g.
CryptoPrimitive_0 =
- CryptoPrimitiveAlgorithmFamiliy
+ CRYPTO_ALGOFAM_3DES
- CryptoPrimitiveAlgorithmMode
+ CRYPTO_ALGOMODE_ECB
- CryptoPrimitiveAlgorithmSecondaryFamiliy
+ CRYPTO_ALGOFAM_NOT_SET
CryptoPrimitive_1 =
- CryptoPrimitiveAlgorithmFamiliy
+ CRYPTO_ALGOFAM_AES
- CryptoPrimitiveAlgorithmMode
+ CRYPTO_ALGOMODE_CBC
- CryptoPrimitiveAlgorithmSecondaryFamiliy
+ CRYPTO_ALGOFAM_NOT_SET
CryptoPrimitive_2 =
- CryptoPrimitiveAlgorithmFamiliy
+ CRYPTO_ALGOFAM_SHA1
- CryptoPrimitiveAlgorithmMode
+ CRYPTO_ALGOMODE_NOT_SET
- CryptoPrimitiveAlgorithmSecondaryFamiliy
+ CRYPTO_ALGOFAM_NOT_SET

CryptoDriverObject
- CryptoPrimitiveRef
+ CryptoPrimitive_0

+ CryptoPrimitive_1
+ CryptoPrimitive_2

Agreed solution:

Extracted from Problem Description:

ECUC_Crypto_00035: Adapt multiplicity to 1..1
ECUC_Crypto_00036: Adapt multiplicity to 1..1
ECUC_Crypto_00037: Adapt multiplicity to 1..1
ECUC_Crypto_00018: Adapt multiplicity to 1..*
–Last change on issue 77304 comment 3–

BW-C-Level:

Application	Specification	Bus
1	4	1

- RfC #77711: [CRYPTO] Csm<Service>AlgorithmFamiliy

  Problem description:

  The name of all configuration parameters CsmHash|MacGenerate|MacVerify|...AlgorithmFamiliy is not correctly written.
  There is an "i" before the "y" in "Family".

  RfC 76783 mentioned this for CsmMacGenerateAlgorithmFamiliy only.

  Agreed solution:

  Change Csm<Service>AlgorithmFamiliy to Csm<Service>AlgorithmFamily in the following ECUCs:
  ECUC_Csm_00038
  ECUC_Csm_00188
  ECUC_Csm_00051
  ECUC_Csm_00182
  ECUC_Csm_00066
  ECUC_Csm_00074
  ECUC_Csm_00082
  ECUC_Csm_00089
  ECUC_Csm_00096
  ECUC_Csm_00105

  SWS_CryptoDriver:
  Change Familiy to Family:
  ECUC_Crypto_00035

ECUC_Crypto_00037
–Last change on issue 77711 comment 8–

BW-C-Level:

Application	Specification	Bus
1	3	1

1.7 Specification Item ECUC_Crypto_00036

Trace References:

none

Content:

Name	CryptoPrimitiveAlgorithmModeCryptoPrimitive.CryptoPrimitiveAlgorithmMode Modein container CryptoPrimitive
Description	Determines the algorithm mode used for the crypto service
Multiplicity	1..* 1
Type	EcucEnumerationParamDef

Range	CRYPTO_ALGOMODE_12ROUNDS	0x0D	Crypto Primitive.CryptoPrimitive Algorithm Mode.CRYPTO_ALGOMODE_12ROUNDS
	CRYPTO_ALGOMODE_20ROUNDS	0x0E	Crypto Primitive.CryptoPrimitive Algorithm Mode.CRYPTO_ALGOMODE_20ROUNDS
	CRYPTO_ALGOMODE_8ROUNDS	0x0C	Crypto Primitive.CryptoPrimitive Algorithm Mode.CRYPTO_ALGOMODE_8ROUNDS
	CRYPTO_ALGOMODE_CBC	0x02	Crypto Primitive.CryptoPrimitive Algorithm Mode.CRYPTO_ALGOMODE_CBC
	CRYPTO_ALGOMODE_CFB	0x03	Crypto Primitive.CryptoPrimitive Algorithm Mode.CRYPTO_ALGOMODE_CFB
	CRYPTO_ALGOMODE_CMAC	0x10	Crypto Primitive.CryptoPrimitive Algorithm Mode.CRYPTO_ALGOMODE_CMAC
	CRYPTO_ALGOMODE_CTR	0x05	Crypto Primitive.CryptoPrimitive Algorithm Mode.CRYPTO_ALGOMODE_CTR
	CRYPTO_ALGOMODE_CTRDRBG	0x12	Crypto Primitive.CryptoPrimitive Algorithm Mode.CRYPTO_ALGOMODE_CTRDRBG
	CRYPTO_ALGOMODE_CUSTOM	0xFF	Crypto Primitive.CryptoPrimitive Algorithm Mode.CRYPTO_ALGOMODE_CUSTOM
	CRYPTO_ALGOMODE_ECB	0x01	Crypto Primitive.CryptoPrimitive Algorithm Mode.CRYPTO_ALGOMODE_ECB
	CRYPTO_ALGOMODE_GCM	0x06	Crypto Primitive.CryptoPrimitive Algorithm Mode.CRYPTO_ALGOMODE_GCM
	CRYPTO_ALGOMODE_GMAC	0x11	Crypto Primitive.CryptoPrimitive Algorithm Mode.CRYPTO_ALGOMODE_GMAC
	CRYPTO_ALGOMODE_HMAC	0x0F	Crypto Primitive.CryptoPrimitive Algorithm Mode.CRYPTO_ALGOMODE_HMAC
	CRYPTO_ALGOMODE_NOT_SET	0x00	Crypto Primitive.CryptoPrimitive Algorithm Mode.CRYPTO_ALGOMODE_NOT_SET
	CRYPTO_ALGOMODE_OFB	0x04	Crypto Primitive.CryptoPrimitive Algorithm Mode.CRYPTO_ALGOMODE_OFB
	CRYPTO_ALGOMODE_RSAES_OAEP	[illegible]	Crypto Primitive.CryptoPrimitive

Post-Build Variant Value	false		
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76744: Differences of Crypto_AlgorithmModeType between CSM and CRYPTO Driver Specifications

  Problem description:

  In CRYPTO Driver spefication, there is no definition about following two values in CryptoPrimitiveAlgorithmMode[ECUC_Crypto_00036]:

  1. CRYPTO_ALGOMODE_SIPHASH_2_4
  2. CRYPTO_ALGOMODE_SIPHASH_4_8

  However, in CSM specification, mentioned above two values are provided in Crypto_AlgorithmModeType [SWS_Csm_01048]:

  1. CRYPTO_ALGOMODE_SIPHASH_2_4 0x13
  2. CRYPTO_ALGOMODE_SIPHASH_4_8 0x14
  –Last change on issue 76744 comment 14–

  Agreed solution:

  add two definitions at [ECUC_Crypto_00036] to Range:

  CRYPTO_ALGOMODE_SIPHASH_2_4 0x13
  CRYPTO_ALGOMODE_SIPHASH_4_8 0x14
  –Last change on issue 76744 comment 8–

  BW-C-Level:

Application	Specification	Bus
1	4	1

- RfC #77304: [CRYPTO] Specification of configurations of CryptoPrimitive (ECUC_Crypto_00033) is not sensible

  Problem description:

Container CryptoPrimitive (ECUC_Crypto_00033) contains
- 1..1 parameters CryptoPrimitiveService
- 1..* parameters CryptoPrimitiveAlgorithmFamiliy
- 1..* parameters CryptoPrimitiveAlgorithmMode
- 1..* parameters CryptoPrimitiveAlgorithmSecondaryFamiliy

This results in incoherent sets of algorithm families and modes per CryptoPrimitive and an improper service very probably.
How shall be determined (by the CryIf) which modes are valid for which familiy, especially if there are several families each supporting another specific mode of the same group of modes.
Why is CryptoPrimitiveService necessary?

E.g.
CryptoPrimitive =
- CryptoPrimitiveService
- MAC_GENERATE
- CryptoPrimitiveAlgorithmFamiliy
+ CRYPTO_ALGOFAM_3DES
+ CRYPTO_ALGOFAM_AES
+ CRYPTO_ALGOFAM_SHA1
- CryptoPrimitiveAlgorithmMode
+ CRYPTO_ALGOMODE_ECB (<– only valid for CRYPTO_ALGOFAM_3DES)
+ CRYPTO_ALGOMODE_CBC (<– only valid for CRYPTO_ALGOFAM_AES)
+ CRYPTO_ALGOMODE_XTS
+ CRYPTO_ALGOMODE_RSASSA_PSS
- CryptoPrimitiveAlgorithmSecondaryFamiliy
+ CRYPTO_ALGOFAM_NOT_SET

On the other hand this is not a "Configuration of a [clear] CryptoPrimitive" as the description of ECUC_Crypto_00033 specifies, but a set of primitives provided by a CryptoDriverObject.

Proposal:
CryptoPrimitive contains
- 1..1 parameters CryptoPrimitiveAlgorithmFamiliy
- 1..1 parameters CryptoPrimitiveAlgorithmMode
- 1..1 parameters CryptoPrimitiveAlgorithmSecondaryFamiliy

CryptoDriverObject contains
- 1..* parameters CryptoPrimitiveRef (or 1..* containers CryptoPrimitiveRefContainer with single parameter 1..1 CryptoPrimitiveRef)

E.g.
CryptoPrimitive_0 =
- CryptoPrimitiveAlgorithmFamiliy
+ CRYPTO_ALGOFAM_3DES
- CryptoPrimitiveAlgorithmMode
+ CRYPTO_ALGOMODE_ECB
- CryptoPrimitiveAlgorithmSecondaryFamiliy
+ CRYPTO_ALGOFAM_NOT_SET
CryptoPrimitive_1 =
- CryptoPrimitiveAlgorithmFamiliy
+ CRYPTO_ALGOFAM_AES
- CryptoPrimitiveAlgorithmMode
+ CRYPTO_ALGOMODE_CBC
- CryptoPrimitiveAlgorithmSecondaryFamiliy
+ CRYPTO_ALGOFAM_NOT_SET
CryptoPrimitive_2 =
- CryptoPrimitiveAlgorithmFamiliy
+ CRYPTO_ALGOFAM_SHA1
- CryptoPrimitiveAlgorithmMode
+ CRYPTO_ALGOMODE_NOT_SET
- CryptoPrimitiveAlgorithmSecondaryFamiliy
+ CRYPTO_ALGOFAM_NOT_SET

CryptoDriverObject
- CryptoPrimitiveRef
+ CryptoPrimitive_0
+ CryptoPrimitive_1
+ CryptoPrimitive_2

Agreed solution:

Extracted from Problem Description:

ECUC_Crypto_00035: Adapt multiplicity to 1..1
ECUC_Crypto_00036: Adapt multiplicity to 1..1
ECUC_Crypto_00037: Adapt multiplicity to 1..1
ECUC_Crypto_00018: Adapt multiplicity to 1..*
–Last change on issue 77304 comment 3–

BW-C-Level:

Application	Specification	Bus
1	4	1

1.8 Specification Item ECUC_Crypto_00037

Trace References:

none

Content:

Name	CryptoPrimitiveAlgorithmSecondaryFamiliyFamilyCryptoPrimitive.CryptoPrimitiveAlgorithm SecondaryFamiliyin container Family
Parent Container	CryptoPrimitive
Description	Determines the algorithm secondary family used for the crypto service
Multiplicity	1..* 1
Type	EcucEnumerationParamDef

Range	CRYPTO_ALGOFAM_3DES	0x13	Crypto Primitive.CryptoPrimitive AlgorithmSecondary FamiliyFamily.CRYPTO_ALGOFAM_3DES
	CRYPTO_ALGOFAM_AES	0x14	Crypto Primitive.CryptoPrimitive AlgorithmSecondary FamiliyFamily.CRYPTO_ALGOFAM_AES
	CRYPTO_ALGOFAM_BLAKE_1_256	0x0F	Crypto Primitive.CryptoPrimitive AlgorithmSecondary FamiliyFamily.CRYPTO_ALGOFAM_BLAKE_1_256
	CRYPTO_ALGOFAM_BLAKE_1_512	0x10	Crypto Primitive.CryptoPrimitive AlgorithmSecondary FamiliyFamily.CRYPTO_ALGOFAM_BLAKE_1_512
	CRYPTO_ALGOFAM_BLAKE_2s_256	0x11	Crypto Primitive.CryptoPrimitive AlgorithmSecondary FamiliyFamily.CRYPTO_ALGOFAM_BLAKE_2s_256
	CRYPTO_ALGOFAM_BLAKE_2s_512	0x12	Crypto Primitive.CryptoPrimitive AlgorithmSecondary FamiliyFamily.CRYPTO_ALGOFAM_BLAKE_2s_512
	CRYPTO_ALGOFAM_BRAINPOOL	0x18	Crypto Primitive.CryptoPrimitive AlgorithmSecondary FamiliyFamily.CRYPTO_ALGOFAM_BRAINPOOL
	CRYPTO_ALGOFAM_CHACHA	0x15	Crypto Primitive.CryptoPrimitive AlgorithmSecondary FamiliyFamily.CRYPTO_ALGOFAM_CHACHA
	CRYPTO_ALGOFAM_CUSTOM	0xFF	Crypto Primitive.CryptoPrimitive AlgorithmSecondary FamiliyFamily.CRYPTO_ALGOFAM_CUSTOM
	CRYPTO_ALGOFAM_ECCNIST	0x19	Crypto Primitive.CryptoPrimitive AlgorithmSecondary FamiliyFamily.CRYPTO_ALGOFAM_ECCNIST
	CRYPTO_ALGOFAM_ECIES	0x1D	Crypto Primitive.CryptoPrimitive AlgorithmSecondary FamiliyFamily.CRYPTO_ALGOFAM_ECIES
	CRYPTO_ALGOFAM_ED25519	0x17	Crypto Primitive.CryptoPrimitive AlgorithmSecondary FamiliyFamily.CRYPTO_ALGOFAM_ED25519
	CRYPTO_ALGOFAM_NOT_SET	0x00	Crypto Primitive.CryptoPrimitive AlgorithmSecondary FamiliyFamily.CRYPTO_ALGOFAM_NOT_SET
	CRYPTO_ALGOFAM_RIPEMD160	0x0E	Crypto Primitive.CryptoPrimitive AlgorithmSecondary FamiliyFamily.CRYPTO_ALGOFAM_RIPEMD160
	CRYPTO_ALGOFAM_RNG	0x1B	Crypto Primitive.CryptoPrimitive AlgorithmSecondary FamiliyFamily.CRYPTO_ALGOFAM_RNG
	CRYPTO_ALGOFAM_RSA	0x16	Crypto

Post-Build Variant Value	false		
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77304: [CRYPTO] Specification of configurations of CryptoPrimitive (ECUC_Crypto_00033) is not sensible

  Problem description:

  Container CryptoPrimitive (ECUC_Crypto_00033) contains
  - 1..1 parameters CryptoPrimitiveService
  - 1..* parameters CryptoPrimitiveAlgorithmFamiliy
  - 1..* parameters CryptoPrimitiveAlgorithmMode
  - 1..* parameters CryptoPrimitiveAlgorithmSecondaryFamiliy
  This results in incoherent sets of algorithm families and modes per CryptoPrimitive and an improper service very probably.
  How shall be determined (by the CryIf) which modes are valid for which familiy, especially if there are several families each supporting another specific mode of the same group of modes.
  Why is CryptoPrimitiveService necessary?

  E.g.
  CryptoPrimitive =
  - CryptoPrimitiveService
  - MAC_GENERATE
  - CryptoPrimitiveAlgorithmFamiliy
  + CRYPTO_ALGOFAM_3DES
  + CRYPTO_ALGOFAM_AES
  + CRYPTO_ALGOFAM_SHA1
  - CryptoPrimitiveAlgorithmMode
  + CRYPTO_ALGOMODE_ECB (<– only valid for CRYPTO_ALGOFAM_3DES)
  + CRYPTO_ALGOMODE_CBC (<– only valid for CRYPTO_ALGOFAM_AES)
  + CRYPTO_ALGOMODE_XTS
  + CRYPTO_ALGOMODE_RSASSA_PSS
  - CryptoPrimitiveAlgorithmSecondaryFamiliy
  + CRYPTO_ALGOFAM_NOT_SET

On the other hand this is not a "Configuration of a [clear] CryptoPrimitive" as the description of ECUC_Crypto_00033 specifies, but a set of primitives provided by a CryptoDriverObject.

Proposal:
CryptoPrimitive contains
- 1..1 parameters CryptoPrimitiveAlgorithmFamiliy
- 1..1 parameters CryptoPrimitiveAlgorithmMode
- 1..1 parameters CryptoPrimitiveAlgorithmSecondaryFamiliy
CryptoDriverObject contains
- 1..* parameters CryptoPrimitiveRef (or 1..* containers CryptoPrimitiveRefContainer with single parameter 1..1 CryptoPrimitiveRef)

E.g.
CryptoPrimitive_0 =
- CryptoPrimitiveAlgorithmFamiliy
+ CRYPTO_ALGOFAM_3DES
- CryptoPrimitiveAlgorithmMode
+ CRYPTO_ALGOMODE_ECB
- CryptoPrimitiveAlgorithmSecondaryFamiliy
+ CRYPTO_ALGOFAM_NOT_SET
CryptoPrimitive_1 =
- CryptoPrimitiveAlgorithmFamiliy
+ CRYPTO_ALGOFAM_AES
- CryptoPrimitiveAlgorithmMode
+ CRYPTO_ALGOMODE_CBC
- CryptoPrimitiveAlgorithmSecondaryFamiliy
+ CRYPTO_ALGOFAM_NOT_SET
CryptoPrimitive_2 =
- CryptoPrimitiveAlgorithmFamiliy
+ CRYPTO_ALGOFAM_SHA1
- CryptoPrimitiveAlgorithmMode
+ CRYPTO_ALGOMODE_NOT_SET
- CryptoPrimitiveAlgorithmSecondaryFamiliy
+ CRYPTO_ALGOFAM_NOT_SET

CryptoDriverObject
- CryptoPrimitiveRef
+ CryptoPrimitive_0

+ CryptoPrimitive_1
+ CryptoPrimitive_2

Agreed solution:

Extracted from Problem Description:

ECUC_Crypto_00035: Adapt multiplicity to 1..1
ECUC_Crypto_00036: Adapt multiplicity to 1..1
ECUC_Crypto_00037: Adapt multiplicity to 1..1
ECUC_Crypto_00018: Adapt multiplicity to 1..*
–Last change on issue 77304 comment 3–

BW-C-Level:

Application	**Specification**	**Bus**
1	4	1

- RfC #77711: [CRYPTO] Csm<Service>AlgorithmFamiliy

  Problem description:

  The name of all configuration parameters CsmHash|MacGenerate|MacVerify|...AlgorithmFamiliy is not correctly written.
  There is an "i" before the "y" in "Family".

  RfC 76783 mentioned this for CsmMacGenerateAlgorithmFamiliy only.

  Agreed solution:

  Change Csm<Service>AlgorithmFamiliy to Csm<Service>AlgorithmFamily in the following ECUCs:
  ECUC_Csm_00038
  ECUC_Csm_00188
  ECUC_Csm_00051
  ECUC_Csm_00182
  ECUC_Csm_00066
  ECUC_Csm_00074
  ECUC_Csm_00082
  ECUC_Csm_00089
  ECUC_Csm_00096
  ECUC_Csm_00105

  SWS_CryptoDriver:
  Change Familiy to Family:
  ECUC_Crypto_00035

ECUC_Crypto_00037
–Last change on issue 77711 comment 8–

BW-C-Level:

Application	Specification	Bus
1	3	1

1.9 Specification Item ECUC_Crypto_00040

Trace References:

none

Content:

Name	CryptoInstanceIdCryptoGeneral.CryptoInstanceId		
Parent Container	CryptoGeneral		
Description	Instance ID of the crypto driver. This ID is used to discern several crypto drivers in case more than one driver is used in the same ECU.		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 255		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77845: [diverse] Configuration of instance ID for instantiated modules

  Problem description:

  Some modules that can exist multiple times in an AUTOSAR BSW stack have configurable instance IDs that are used to e.g. call DET. Examples are the bus drivers. Others, like the CDD, Crypto driver, or DIO driver, lack such a configuration parameter.
  –Last change on issue 77845 comment 2–

Agreed solution:

TPS EcuConfigurationSpecification (CDD):

Add container CddGeneral with one parameter CddInstanceId to Cdd EcucModuleDef
Description: Specifies the InstanceId of this module instance. If only one instance is present it shall have the Id 0.
Multiplicity: 1
Type: EcucIntegerParamDef
Range: 0 .. 255
Default value: -
Post-Build Variant Value: false
Value Configuration Class: Pre-compile time - All Variants
==
Crypto:

Add new pre-compile integer parameter "CryptoInstanceId" (range 0..255) to the container CryptoGeneral,
Description: "Instance ID of the crypto driver. This ID is used to discern several crypto drivers in case more than one driver is used in the same ECU."
==
Eep:

Change ECUC_Eep_00189 Description from : "Represents the Index of the driver, used by EA" to
"Specifies the InstanceId of this module instance. If only one instance is present it shall have the Id 0."
==
Fls:

Add REQ:
SWS_Fls_xxx: If more than one instance of the flash driver is used in an ECU, the individual instances have to be given a unique instance ID. This instance ID shall be configured as the parameter FlsDriverIndex. If only one instance of the flash driver is used in an ECU, this instance shall have the parameter FlsDriverIndex configured as 0.
==
Wdg:

Change ECUC_Wdg_00117 Description from : "Represents the watchdog driver’s ID so that it can be referenced by the watchdog interface." to
"Specifies the InstanceId of this module instance. If only one instance is present it

shall have the Id 0."
==
Xfrm:

Add into the container XfrmGeneral a new parameter XfrmInstanceId to Xfrm EcucModuleDef
Description: Specifies the InstanceId of this module instance. If only one instance is present it shall have the Id 0.
Multiplicity: 1
Type: EcucIntegerParamDef
Range: 0 .. 255
Default value: -
Post-Build Variant Value: false
Value Configuration Class: Pre-compile time - All Variants
–Last change on issue 77845 comment 32–

BW-C-Level:

Application	Specification	Bus
1	3	1

1.10 Specification Item ECUC_Crypto_00041

Trace References:

none

Content:

Name	CryptoKeyElementFormatCryptoKeyElement.CryptoKeyElementFormat
Parent Container	CryptoKeyElement
Description	Defines the format for the key element. This is the format used to provide or extract the key data from the driver.
Multiplicity	1
Type	EcucEnumerationParamDef

Range	CRYPTO_KE_FORMAT_BIN_CERT_CVC	0x08	CryptoKeyElement.CryptoKeyElementFormat.CRYPTO_KE_FORMAT_BIN_CERT_CVC
	CRYPTO_KE_FORMAT_BIN_CERT_X509_V3	0x07	CryptoKeyElement.CryptoKeyElementFormat.CRYPTO_KE_FORMAT_BIN_CERT_X509_V3
	CRYPTO_KE_FORMAT_BIN_IDENT_PRIVATEKEY_PKCS8	0x03	CryptoKeyElement.CryptoKeyElementFormat.CRYPTO_KE_FORMAT_BIN_IDENT_PRIVATEKEY_PKCS8
	CRYPTO_KE_FORMAT_BIN_IDENT_PUBLICKEY	0x04	CryptoKeyElement.CryptoKeyElementFormat.CRYPTO_KE_FORMAT_BIN_IDENT_PUBLICKEY
	CRYPTO_KE_FORMAT_BIN_OCTET	0x01	CryptoKeyElement.CryptoKeyElementFormat.CRYPTO_KE_FORMAT_BIN_OCTET
	CRYPTO_KE_FORMAT_BIN_RSA_PRIVATEKEY	0x05	CryptoKeyElement.CryptoKeyElementFormat.CRYPTO_KE_FORMAT_BIN_RSA_PRIVATEKEY
	CRYPTO_KE_FORMAT_BIN_RSA_PUBLICKEY	0x06	CryptoKeyElement.CryptoKeyElementFormat.CRYPTO_KE_FORMAT_BIN_RSA_PUBLICKEY
	CRYPTO_KE_FORMAT_BIN_SHEKEYS	0x02	CryptoKeyElement.CryptoKeyElementFormat.CRYPTO_KE_FORMAT_BIN_SHEKEYS
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency			

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77661: Definition for asymmetric key formats

  Problem description:

  Name: Armin Happel

  Description/Motivation:
  Currently, the AUTOSAR crypto stack specifies to provide asymmetric key material in PKCS# 8 format only [see SWS_CSM_00951]. However, the standard is not precise enough and defines only the usage of private key material. Optionally, public key material can be provided in addition. This provides the lack of definition in the AUTOSAR stack, that public keys cannot be provided for certain algorithms, such as signature verification.
  This RFC extends the current definition so that also public key material can be provided to the crypto stack.

  Agreed solution:

  See attachment: https://bugzilla.autosar.org/attachment.cgi?id=4617
  –Last change on issue 77661 comment 29–

  BW-C-Level:

Application	Specification	Bus
1	4	1

1.11 Specification Item SWS_Crypto_00037

Trace References:

none

Content:

The index of the different key elements from the different crypto services are defined as in imported types table SWS_Crypto_00044. Csm_01022.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76782: [CRYPTO] Missing Information about SWS_Crypto_00044

  Problem description:

According to [SWS_Crypto_00037], the index of the key elements are defined in imported types table [SWS_Crypto_00044].
However, there is no description about [SWS_Crypto_00044] in CRYPTO specification.

The following description may apply to [SWS_Crypto_00044], but the description is broken.

The Crypto Stack API uses the key element index definition from the CSM module.
Type definitions
N/A.

Could you please check and adjust it?

Agreed solution:

CryptoDriver:
- SWS_Crypto_00037: Replace SWS_Crypto_00044 by SWS_Csm_01022

CSM
- SWS_Csm_01022: Correct the tag by moving it out of the table
–Last change on issue 76782 comment 10–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.12 Specification Item SWS_Crypto_00039

Trace References:

none

Content:

If a key is in the state "invalid", crypto services which make use of that key, shall return with CRYPTO_E_KEY_INVALIDNOT_VALID.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76827: [CRYPTO] Which API returns CRYPTO_E_KEY_INVALID?

  Problem description:

In [SWS_Crypto_00039], it is mentioned if a key is in the state "invalid", crypto services shall return CRYPTO_E_KEY_INVALID. However, in section 8.2 Function definitions, there are no definitions which API returns CRYPTO_E_KEY_INVALID.

Problem 1) According to our understanding, in CYRYPTO specification, it should be mentioned which API returns CRYPTO_KEY_INAVALID. Could you please mention these solutions in section 8.2 Function definition?

Problem 2) In addition, there is no defintion about value of CRYPTO_E_KEY_INVALID in [SWS_Crypto_00043]. Could you please define the value of CRYPTO_E_KEY_INVALID?

Agreed solution:

In [SWS_Crypto_00039]:
Replace "CRYPTO_E_KEY_INVALID" with "CRYPTO_E_KEY_NOT_VALID".
–Last change on issue 76827 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.13 Specification Item SWS_Crypto_00047

Trace References:

none

Content:

If the parameter versioninfo is a null pointer and if default development error detection for the Crypto Driver is enabled, the function Crypto_GetVersionInfo shall report CRYPTO_E_PARAM_POINTER to the DET.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

SWS_CryIf:
replace "default error" detection with "development error detection" in requirement:
[SWS_CryIf_00016]
[SWS_CryIf_00017]
[SWS_CryIf_00027]
[SWS_CryIf_00028]
[SWS_CryIf_00029]
[SWS_CryIf_00129]
[SWS_CryIf_00130]
[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]

[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]

[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer

–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.14 Specification Item SWS_Crypto_00057

Trace References:

none

Content:

If the module is not initialized and if default development error detection for the Crypto Driver is enabled, the function Crypto_ProcessJob shall report CRYPTO_E_UNINIT to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]

[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:

replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]

[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.15 Specification Item SWS_Crypto_00058

Trace References:

none

Content:

If the parameter objectId is out of range and if default development error detection for the Crypto Driver is enabled, the function Crypto_ProcessJob shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]
  [SWS_CryIf_00112]
  [SWS_CryIf_00116]
  [SWS_CryIf_00117]
  [SWS_CryIf_00118]
  [SWS_CryIf_00068]
  [SWS_CryIf_00069]
  [SWS_CryIf_00070]

[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]

[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]

[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.16 Specification Item SWS_Crypto_00059

Trace References:

none

Content:

If the parameter job is a null pointer and if default development error detection for the Crypto Driver is enabled, the function Crypto_ProcessJob shall report CRYPTO_E_PARAM_POINTER to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:

[SWS_CryIf_00016]
[SWS_CryIf_00017]
[SWS_CryIf_00027]
[SWS_CryIf_00028]
[SWS_CryIf_00029]
[SWS_CryIf_00129]
[SWS_CryIf_00130]
[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]

[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]

[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.17 Specification Item SWS_Crypto_00064

Trace References:

none

Content:

If the parameter job->jobPrimitiveInfo->primitiveInfo->service is not supported by the Crypto Driver Object and

if default development error detection for the Crypto Driver is enabled, the function Crypto_ProcessJob shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]

[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:

replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]

[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.18 Specification Item SWS_Crypto_00067

Trace References:

none

Content:

If the parameter job->jobPrimitiveInfo->primitiveInfo->algorithm (with its variation in family, keyLength and mode) is not supported by the Crypto Driver Object and if default development error detection for the Crypto Driver is enabled, the function Crypto_ProcessJob shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]
  [SWS_CryIf_00112]
  [SWS_CryIf_00116]
  [SWS_CryIf_00117]
  [SWS_CryIf_00118]
  [SWS_CryIf_00068]
  [SWS_CryIf_00069]

[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]

[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]

[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.19 Specification Item SWS_Crypto_00070

Trace References:

none

Content:

If a pointer is required as an argument, but it is a null pointer, the Crypto_ProcessJob() function shall report CRYPTO_E_PARAM_POINTER. If the value, which is pointed by a length pointer, is zero, and if default development error detection for the Crypto Driver is enabled, the Crypto_ProcessJob() function report CRYPTO_E_PARAM_VALUE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

Agreed solution:

SWS_CryIf:
replace "default error" detection with "development error detection" in requirement:
[SWS_CryIf_00016]
[SWS_CryIf_00017]
[SWS_CryIf_00027]
[SWS_CryIf_00028]
[SWS_CryIf_00029]
[SWS_CryIf_00129]
[SWS_CryIf_00130]
[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]

[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]

[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:

[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.20 Specification Item SWS_Crypto_00071

Trace References:

none

Content:

Member Service*	input Ptr	input Length Ptr	secondary Input Ptr	secondary Input Length Ptr	tertiary Input Ptr	tertiary Input Length Ptr	output Ptr	output Length Ptr	secondary Output Ptr	secondary Output Length Ptr	verify Ptr	output64Ptr	mode
HASH	UG	UG					F	F					SUF
MACGENERATE	UG	UG					F	F					SUF
MACVERIFY	UG	UG	F	F							F		SUF
ENCRYPT	UG	UG					UF	UF					SUF
DECRYPT	UG	UG					UF	UF					SUF
AEADENCRYPT	UG	UG	F	F			UF	UF	F	F			SUF
AEADDECRYPT	UG	UG	F	F	F	F	UF	UF			F		SUF
SIGNATUREGENERATE	UG	UG					F	F					SUF
SIGNATUREVERIFY	UG	UG	F	F							F		SUF
SECCOUNTERINCREMENT													
SECCOUNTERREAD												F	
RANDOMGENERATE							F	F					

*: Service names are derived from Crypto_ServiceInfoType (part of job struct)

S: member required in Start mode.

U: member required in Update mode.

F: member required in Finish mode.

G: member optional in Finish mode.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76783: Typo or copy/paste mistakes

  Problem description:

  Hello,

  I found some other mistakes in the specification documents. Most of them are typos or copy/paste mistakes. As document owner of the CryptoServiceManager, I need a confirmation from someone else, before I can implement them into the document.

  AUTOSAR_SWS_CryptoDriver:
  [SWS_Crypto_00139]: CRYPTO_E_KEY_EXTRACT_DENIED does not exist anymore. Replace error code with CRYPTO_E_KEY_READ_FAIL.
  [SWS_Crypto_91005]: Crypto_KeyValidSet() shall be named analogously to Csm_KeySetValid() and CryIf_KeySetValid(). Therefore, rename Crypto_KeyValidSet() to Crypto_KeySetValid().
  [SWS_Crypto_00071]: In table: inputLengthPtr, secondaryInputLengthPtr, tertiaryInputLengthPtr are no pointer anymore. rename them to inputLength, secondaryInputLength, tertiaryInputLength

  AUTOSAR_SWS_CryptoServiceManager:
  [SWS_Csm_01035]: Csm_KeyCopy() shall call CryIf_KeyCopy() not CryIf_KeyElementCopy().
  [SWS_Csm_01080]: Csm_AsymPrivateKeyType is not up-to-date. It should be modified like [SWS_Csm_00076] Csm_AsymPublicKeyType or [SWS_Csm_01082] Csm_SymKeyType.
  SWS_Csm_00455
  [SWS_Csm_00455]: tag as obsolete
  [ECUC_Csm_00188]: typo: CsmMacGenerateAlgorithmFamiliy -> CsmMacGenerateAlgorithmFamily
  [ECUC_Csm_00049]: CsmMacVerifyAlgorithmMode missing. (see analogues CsmMacGenerateAlgorithmMode [ECUC_Csm_00189])
  [SWS_Csm_00966]: CopyPaste mistake: Delete: "Wrong return values - here are the correct ones:"
  [SWS_Csm_01023]: plaintextLength description wrong. replace with "Contains the number of bytes to encrypt."
  [SWS_Csm_01023]: typo "associtatedDataLengthPtr" and it is no pointer. replace with: "associatedDataLength"
  [SWS_Csm_01025]: typo, replace line "job->jobPrimitiveInputOutput.outputLength = ciphertextLength," with
  "job->jobPrimitiveInputOutput.outputLengthPtr = ciphertextLengthPtr,"
  [SWS_Csm_01013]: typo: rename "PrimitiveInputOutput" to "jobPrimitiveInputOut-

put". Or is this rename intended?
Then every assignment of "jobPrimitiveInputOutput" has to be renamed to "primitiveInputOutput" like
"job->jobPrimitiveInputOutput.mode = mode," has to be modified to "job->primitiveInputOutput.mode = mode,"
[SWS_Csm_01026]: typo: replace "associtatedDataLength" with "associatedDataLength"
[SWS_Csm_01027]: missing line: "job->jobPrimitiveInputOutput.verifyPtr = verifyPtr."
[SWS_Csm_00992]: copypaste mistake: replace "mode: The Crypto_JobInfoType job with the corresponding jobId shall be modified in the following way:" with ""mode: Indicates which operation mode(s) to perfom."
[SWS_Csm_00992]: copypaste mistake: replace "resultLengthPtr: Contains the number of bytes of the associated data." with ""resultLengthPtr: Holds a pointer to the memory location in which the output length in bytes of the signature is stored. On calling this function, this parameter shall contain the size of the buffer provided by resultPtr. When the request has finished, the actual length of the returned value shall be stored."
[SWS_Csm_01543]: description wrong. replace with "Generate a random number and stores it in the memory location pointed by the result pointer."
[SWS_Csm_00168]: description wrong, there is no IV. replace with "This function is deprecated. Sets the key for symmetrical encryption."
[SWS_Csm_01031]: description wrong, it is not decrement. "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterRead Service"

A proposed solution is added, too.

Agreed solution:

AUTOSAR_SWS_CryptoDriver:
[SWS_Crypto_00139]: Replace CRYPTO_E_KEY_EXTRACT_DENIED with CRYPTO_E_KEY_READ_FAIL.
[SWS_CryIf_91015]: Remove CRYPTO_E_KEY_EXTRACT_DENIED
[SWS_Crypto_91005]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add Crypto_KeySetValid as API (Description according to SWS_Crypto_91005)
[SWS_Crypto_00082]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add E_UNINIT DET check SWS for Crypto_KeySetValid (Text according to SWS_Crypto_00082)
[SWS_Crypto_00083]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add E_PARAM_HANDLE DET check SWS for Crypto_KeySetValid (Text according to SWS_Crypto_00083)
last sentence in 8.2.4.1.2: Rename Crypto_KeyValidSet to Crypto_KeySetValid

[SWS_Crypto_00071]: rename inputLengthPtr, secondaryInputLengthPtr, tertiaryInputLengthPtr to inputLength, secondaryInputLength, tertiaryInputLength

AUTOSAR_SWS_CryptoServiceManager:
[SWS_Csm_01035]: CryIf_KeyElementCopy() shall be replaced with CryIf_KeyCopy().

[SWS_Csm_01080]: replace with (see [SWS_Csm_00076]):
Name: Csm_AsymPrivateKeyType
Kind: Structure
Elements:
length: uint32: This element contains the length in bytes of the key stored in element 'data'
data: Csm_AsymPrivateKeyArrayType: This element contains the key data or a key handle.
Description: Structure for the private asymmetrical key.
Variation: –

[SWS_Csm_00455]: tag as obsolete

[ECUC_Csm_00188]: typo: CsmMacGenerateAlgorithmFamiliy -> CsmMacGenerateAlgorithmFamily
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmMode (see analogues CsmMacGenerateAlgorithmMode [ECUC_Csm_00189])
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmModeCustom (see analogues CsmMacGenerateAlgorithmModeCustom [ECUC_Csm_00189])
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmKeyLength (see analogues CsmMacGenerateAlgorithmKeyLength [ECUC_Csm_00189])
[SWS_Csm_00966]: Delete: "Wrong return values - here are the correct ones:"
[SWS_Csm_01023]: Replace description with: "Contains the number of bytes to encrypt."
[SWS_Csm_01023]: Replace "associatedDataLengthPtr" with "associatedDataLength"
[SWS_Csm_01025]: Replace line "job->jobPrimitiveInputOutput.outputLength = ciphertextLength," with
"job->jobPrimitiveInputOutput.outputLengthPtr = ciphertextLengthPtr,"
[SWS_Csm_01013]: rename "PrimitiveInputOutput" to "jobPrimitiveInputOutput". rename "state" to "jobState".
[SWS_Csm_01026]: replace "associtatedDataLength" with "associatedDataLength"
[SWS_Csm_01027]: add line: "job->jobPrimitiveInputOutput.verifyPtr = verifyPtr."
[SWS_Csm_00992]: replace "mode: The Crypto_JobInfoType job with the corresponding jobId shall be modified in the following way:" with ""mode: Indicates which operation mode(s) to perform."

[SWS_Csm_00992]: replace "resultLengthPtr: Contains the number of bytes of the associated data." with ""resultLengthPtr: Holds a pointer to the memory location in which the output length in bytes of the signature is stored. On calling this function, this parameter shall contain the size of the buffer provided by resultPtr. When the request has finished, the actual length of the returned value shall be stored."
[SWS_Csm_01543]: replace description with "Generate a random number and stores it in the memory location pointed by the result pointer."
[SWS_Csm_00168]: replace description with "This function is deprecated. Sets the key for symmetrical encryption."
[SWS_Csm_01031]: replace "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterDecrement Service" with "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterRead Service"
–Last change on issue 76783 comment 29–

BW-C-Level:

Application	**Specification**	**Bus**
4	3	1

1.21 Specification Item SWS_Crypto_00073

Trace References:

none

Content:

In the following table the content of the different input and output buffers of job.jobPrimitive InputOutputType are specified:

Parameter Service*	Input	Secondary Input	Tertiary Input	Output	Secondary Output	Input 64	Output 64 Ptr
HASH	plaintext			generated hash			
MACGENERATE	plaintext			generated MAC			
MACVERIFY	plaintext	MAC to be verified					
ENCRYPT	plaintext			encrypted ciphertext			
DECRYPT	ciphertext			decrypted plaintext			
AEADENCRYPT	plaintext	associated Data		encrypted ciphertext	generated Tag		
AEADDECRYPT	ciphertext	associated Data	Tag to be verified	decrypted Plaintext			

Parameter Service*	Input	Secondary Input	Tertiary Input	Output	Secondary Output	Input 64	Output 64 Ptr
SIGNATUREGENERATE	plaintext			generated signature			
SIGNATUREVERIFY	plaintext	signature to be verified					
SECURECOUNTER-INCREMENT						Step size	
SECURECOUNTERREAD							Value of counter
RANDOMGENERATE				random number Generated random			

*: Service names are derived from Crypto_ServiceInfoType.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76745: Missing three CRYIF Interfaces

  Problem description:

  There are no three CRYIF interfaces which are provided for Crypto Service Manager(CSM).
  The CSM specification is described as below:

  1. [SWS_Csm_00973] If no errors are detected by Csm, the service Csm_SecureCounterIncrement() shall call CryIf_SecureCounterIncrement().
  2. [SWS_Csm_01000] If no errors are detected by Csm, the service Csm_SecureCounterRead() shall call CryIf_SecureCounterRead().
  3. [SWS_Csm_01001] The Crypto_JobInfoType job with the corresponding jobId shall be used as parameter in CryIf_RandomGenerate()...

  However, there are no definition of following three CRYIF intefaces in CRYIF specification:

  1. CryIf_SecureCounterIncrement
  2. CryIf_SecureCounterRead
  3. CryIf_RandomGenerate

  Could you please check and solve it?

  Agreed solution:

[SWS_Csm_01009]: Add additional element (after verifyPtr): "input64 uint64 versatile input parameter"

add note to 7.2.2.2.1 after [SWS_Csm_00939]:
Note: The Csm_<Service>() will call the CryIf_ProcessJob() with a pointer to Crypto_JobType, where all the necessary information are stored to process the job. Part of this Crypto_JobType is a Crypto_JobPrimitiveInputOutputType, where all the information about the input and output parameters depending of the service are stored. A definition of the mapping from the API parameters of Csm_<Service>() to the parameters of Crypto_JobPrimitiveInputOutputType, can be found in [SWS_Crypto_00073] of the Crypto Driver specification.

remove the following requirements:
[SWS_Csm_01015]
[SWS_Csm_01017]
[SWS_Csm_01016]
[SWS_Csm_00986]
[SWS_Csm_00990]
[SWS_Csm_01025]
[SWS_Csm_01027]
[SWS_Csm_00993]
[SWS_Csm_00997]
[SWS_Csm_00973]
[SWS_Csm_01000]
[SWS_Csm_01001]

[SWS_Crypto_00073]:
Add to the table the following rows and columns (input64 and output64Ptr are new columns)

Service: Output input64 output64Ptr
SECURECOUNTERINCREMENT step size

SECURECOUNTERREAD value of counter

RANDOMGENERATE generated random

for clarification (Tabulator are not precise enough):
step size should be in column input64
value of counter should be in column output64Ptr
generated random should be in column Output
–Last change on issue 76745 comment 20–

BW-C-Level:

Application	Specification	Bus
4	4	1

1.22 Specification Item SWS_Crypto_00075

Trace References:

none

Content:

If the Crypto Driver is not yet initialized and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyElementSet shall report CRYPTO_E_UNINIT to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]

[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]

[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]

[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.23 Specification Item SWS_Crypto_00076

Trace References:

none

Content:

If cryptoKeyId is out of range and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyElementSet shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]
  [SWS_CryIf_00112]
  [SWS_CryIf_00116]
  [SWS_CryIf_00117]
  [SWS_CryIf_00118]
  [SWS_CryIf_00068]
  [SWS_CryIf_00069]
  [SWS_CryIf_00070]

[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]

[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]

[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.24 Specification Item SWS_Crypto_00077

Trace References:

none

Content:

If parameter keyElementId is out of range and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyElementSet shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:

[SWS_CryIf_00016]
[SWS_CryIf_00017]
[SWS_CryIf_00027]
[SWS_CryIf_00028]
[SWS_CryIf_00029]
[SWS_CryIf_00129]
[SWS_CryIf_00130]
[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]

[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]

[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.25 Specification Item SWS_Crypto_00078

Trace References:

none

Content:

If the parameter keyPtr is a null pointer and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyElementSet shall report CRYPTO_E_PARAM_POINTER to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]

[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]

[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]

[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.26 Specification Item SWS_Crypto_00079

Trace References:

none

Content:

If keyLength is zero and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyElementSet shall report CRYPTO_E_PARAM_VALUE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]
  [SWS_CryIf_00112]
  [SWS_CryIf_00116]
  [SWS_CryIf_00117]
  [SWS_CryIf_00118]
  [SWS_CryIf_00068]
  [SWS_CryIf_00069]
  [SWS_CryIf_00070]

[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]

[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]

[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.27 Specification Item SWS_Crypto_00082

Trace References:

none

Content:

If the module is not yet initialized and default development error detection for the Crypto Driver is enabled, the function Crypto_KeyValidSet shall report CRYPTO_E_UNINIT to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:

[SWS_CryIf_00016]
[SWS_CryIf_00017]
[SWS_CryIf_00027]
[SWS_CryIf_00028]
[SWS_CryIf_00029]
[SWS_CryIf_00129]
[SWS_CryIf_00130]
[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]

[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]

[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.28 Specification Item SWS_Crypto_00083

Trace References:

none

Content:

If parameter cryptoKeyId is out of range and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyValidSet shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]

[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]

[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]

[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.29 Specification Item SWS_Crypto_00085

Trace References:

none

Content:

If the module is not yet initialized and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyElementGet shall report CRYPTO_E_UNINIT to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]
  [SWS_CryIf_00112]
  [SWS_CryIf_00116]
  [SWS_CryIf_00117]
  [SWS_CryIf_00118]
  [SWS_CryIf_00068]
  [SWS_CryIf_00069]
  [SWS_CryIf_00070]

[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]

[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]

[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.30 Specification Item SWS_Crypto_00086

Trace References:

none

Content:

If the parameter cryptoKeyId is out of range and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyElementGet shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:

[SWS_CryIf_00016]
[SWS_CryIf_00017]
[SWS_CryIf_00027]
[SWS_CryIf_00028]
[SWS_CryIf_00029]
[SWS_CryIf_00129]
[SWS_CryIf_00130]
[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]

[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]

[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.31 Specification Item SWS_Crypto_00087

Trace References:

none

Content:

If the parameter keyElementId is out of range and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyElementGet shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]

[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]

[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]

[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.32 Specification Item SWS_Crypto_00088

Trace References:

none

Content:

If the parameter resultPtr is a null pointer and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyElementGet shall report CRYPTO_E_PARAM_POINTER the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]
  [SWS_CryIf_00112]
  [SWS_CryIf_00116]
  [SWS_CryIf_00117]
  [SWS_CryIf_00118]
  [SWS_CryIf_00068]
  [SWS_CryIf_00069]
  [SWS_CryIf_00070]

[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]

[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]

[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.33 Specification Item SWS_Crypto_00089

Trace References:

none

Content:

If the parameter resultLengthPtr is a null pointer and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyElementGet shall report CRYPTO_E_PARAM_POINTER to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:

[SWS_CryIf_00016]
[SWS_CryIf_00017]
[SWS_CryIf_00027]
[SWS_CryIf_00028]
[SWS_CryIf_00029]
[SWS_CryIf_00129]
[SWS_CryIf_00130]
[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]

[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]

[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.34 Specification Item SWS_Crypto_00090

Trace References:

none

Content:

If the value, which is pointed by resultLengthPtr is zero and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyElementGet shall report CRYPTO_E_PARAM_VALUE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]

[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]

[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]

[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.35 Specification Item SWS_Crypto_00093

Trace References:

none

Content:

If the buffer resultPtr is too small to store the result of the request, CRYPTO_E_SMALL_BUFFER shall be returned and if default development error detection is enabled, CRYPTO_E_SMALL_BUFFER shall be reported to the DET.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]
  [SWS_CryIf_00112]
  [SWS_CryIf_00116]
  [SWS_CryIf_00117]
  [SWS_CryIf_00118]
  [SWS_CryIf_00068]
  [SWS_CryIf_00069]
  [SWS_CryIf_00070]

[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]

[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]

[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.36 Specification Item SWS_Crypto_00094

Trace References:

none

Content:

If the module is not yet initialized and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyGenerate shall report CRYPTO_E_UNINIT to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:

[SWS_CryIf_00016]
[SWS_CryIf_00017]
[SWS_CryIf_00027]
[SWS_CryIf_00028]
[SWS_CryIf_00029]
[SWS_CryIf_00129]
[SWS_CryIf_00130]
[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]

[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]

[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.37 Specification Item SWS_Crypto_00095

Trace References:

none

Content:

If the parameter cryptoKeyId is out of range and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyGenerate shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]

[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]

[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]

[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.38 Specification Item SWS_Crypto_00097

Trace References:

none

Content:

If the module is not yet initialized and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyDerive shall report CRYPTO_E_UNINIT to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]
  [SWS_CryIf_00112]
  [SWS_CryIf_00116]
  [SWS_CryIf_00117]
  [SWS_CryIf_00118]
  [SWS_CryIf_00068]
  [SWS_CryIf_00069]
  [SWS_CryIf_00070]

[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]

[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]

[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.39 Specification Item SWS_Crypto_00098

Trace References:

none

Content:

If the parameter cryptoKeyId is out of range and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyDerive shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:

[SWS_CryIf_00016]
[SWS_CryIf_00017]
[SWS_CryIf_00027]
[SWS_CryIf_00028]
[SWS_CryIf_00029]
[SWS_CryIf_00129]
[SWS_CryIf_00130]
[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]

[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]

[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.40 Specification Item SWS_Crypto_00103

Trace References:

none

Content:

If the module is not yet initialized and if default development error detection for the Crypto Driver is enabled: The function Crypto_KeyExchangeCalcPubVal shall report CRYPTO_E_UNINIT to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]

[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]

[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]

[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.41 Specification Item SWS_Crypto_00104

Trace References:

none

Content:

If the parameter cryptoKeyId is out of range and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyExchangeCalcPubVal shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]
  [SWS_CryIf_00112]
  [SWS_CryIf_00116]
  [SWS_CryIf_00117]
  [SWS_CryIf_00118]
  [SWS_CryIf_00068]
  [SWS_CryIf_00069]
  [SWS_CryIf_00070]

[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]

[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]

[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.42 Specification Item SWS_Crypto_00105

Trace References:

none

Content:

If the parameter publicValuePtr is a null pointer and if ~~default~~ development error detection for the Crypto Driver is enabled, the function Crypto_KeyExchangeCalcPubVal shall report CRYPTO_E_PARAM_POINTER to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:

[SWS_CryIf_00016]
[SWS_CryIf_00017]
[SWS_CryIf_00027]
[SWS_CryIf_00028]
[SWS_CryIf_00029]
[SWS_CryIf_00129]
[SWS_CryIf_00130]
[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]

[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]

[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.43 Specification Item SWS_Crypto_00106

Trace References:

none

Content:

If the parameter pubValueLengthPtr is a null pointer and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyExchangeCalcPubVal shall report CRYPTO_E_PARAM_POINTER to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]

[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]

[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]

[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.44 Specification Item SWS_Crypto_00107

Trace References:

none

Content:

If the value, which is pointed by pubValueLengthPtr is zero and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyExchangeCalcPubVal shall report CRYPTO_E_PARAM_VALUE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]
  [SWS_CryIf_00112]
  [SWS_CryIf_00116]
  [SWS_CryIf_00117]
  [SWS_CryIf_00118]
  [SWS_CryIf_00068]
  [SWS_CryIf_00069]
  [SWS_CryIf_00070]

[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]

[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]

[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.45 Specification Item SWS_Crypto_00110

Trace References:

none

Content:

If the buffer publicValuePtr is too small to store the result of the request, CRYPTO_E_SMALL_BUFFER shall be returned and if default error detection is enabled, the function shall additionally report the runtime error CRYPTO_E_RE_SMALL_BUFFERshall be reported to the DET.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76636: Rollout of ’Runtime errors’ for entire crypto stack documents

  Problem description:

  Crypto Stack documents are not in line with the RfC # 59085.

  In SWS_secureOnboardCommunication
  Example1: SECOC_E_CRYPTO_FAILURE in the is a development error, but should

be a runtime error.

In SWS_CryptoServiceManager
Example2: CSM_E_SERVICE_NOT_STARTED is not referenced.

Example3: CSM_E_PARAM_HANDLE is not referenced in chapter 7.3. It is not clear development error or runtime error.
–Last change on issue 76636 comment 33–

Agreed solution:

CryptoInterface:
https://bugzilla.autosar.org/attachment.cgi?id=4587

CryptoServiceManager:
https://bugzilla.autosar.org/attachment.cgi?id=4614

CryptoDriver:
https://bugzilla.autosar.org/attachment.cgi?id=4613

SecureOnboardCommunication:
https://bugzilla.autosar.org/attachment.cgi?id=4598
–Last change on issue 76636 comment 41–

BW-C-Level:

Application	Specification	Bus
1	4	1

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]

[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]

[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]

[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.46 Specification Item SWS_Crypto_00111

Trace References:

none

Content:

If the module is not yet initialized and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyExchangeCalcSecret shall report CRYPTO_E_UNINIT to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]

[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]

[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]

[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.47 Specification Item SWS_Crypto_00112

Trace References:

none

Content:

If the parameter cryptoKeyId is out of range and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyExchangeCalcSecret shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

Problem description:

replace "default error detection" with "development error detection"
–Last change on issue 76932 comment 2–

Agreed solution:

SWS_CryIf:
replace "default error" detection with "development error detection" in requirement:
[SWS_CryIf_00016]
[SWS_CryIf_00017]
[SWS_CryIf_00027]
[SWS_CryIf_00028]
[SWS_CryIf_00029]
[SWS_CryIf_00129]
[SWS_CryIf_00130]
[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]

[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]

[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]

[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.48 Specification Item SWS_Crypto_00113

Trace References:

none

Content:

If the parameter partnerPublicValuePtr is a null pointer and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyExchangeCalcSecret shall report CRYPTO_E_PARAM_POINTER to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]

[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]

[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]

[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.49 Specification Item SWS_Crypto_00115

Trace References:

none

Content:

If partnerPubPublicValueLength is zero and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyExchangeCalcSecret shall report CRYPTO_E_PARAM_VALUE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]

[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]

[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]

[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

- RfC #77937: [CRYPTO] Parameters inconsistences in SWS Crypto

  Problem description:

  Hello,
  Please verify the following inconsistencies between naming inside AUTOSAR_SWS_CryptoDriver:

  1.
  [SWS_Crypto_91013] the parameter name is entropyLength
  [SWS_Crypto_00131] the parameter name is seedLength
  Is parameter name seedLength or entropyLength?

  2.
  [SWS_Crypto_91010] the parameter name is partnerPublicValueLength
  [SWS_Crypto_00115] the parameter name is partnerPubValueLength.
  Is parameter name partnerPublicValueLength or partnerPubValueLength?

3.
[SWS_Crypto_00171] the parameter name is verifyCryptoKeyId
[SWS_Crypto_00174] the parameter name is validateCryptoKeyId
Is parameter name verifyCryptoKeyId or validateCryptoKeyId?

Thank you,
Alexandra

Agreed solution:

[SWS_Crypto_00130] Replace seedPtr with entropyPtr
[SWS_Crypto_00131] Replace seedLength with entropyLength.
[SWS_Crypto_00115] Replace partnerPubValueLength with partnerPublicValueLength.
[SWS_Crypto_00174] Replace validateCryptoKeyId with verifyCryptoKeyId
–Last change on issue 77937 comment 3–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.50 Specification Item SWS_Crypto_00120

Trace References:

none

Content:

If the job is synchronous, the function Crypto_ProcessJob() shall wait while the crypto driver object is busy and process the job when the crypto driver object is idle.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77372: Request of synchronous job when crypto driver object is busy

  Problem description:

  There are conflicting requirements about the handling of requests of synchronous jobs when the corresponding crypto driver object is already processing another job:

  [SWS_Crypto_00120] If the job is synchronous, the function Crypto_ProcessJob() shall wait while the crypto driver object is busy and process the job when the crypto driver object is idle.

Is in opposition to:
[SWS_Crypto_00034] If Crypto_ProcessJob() is called with synchronous job processing and the queue is not full, but the Crypto Driver Object is busy, the Crypto Driver Object shall not queue the job and return CRYPTO_E_BUSY. No job shall be put in any queue.

If Crypto_ProcessJob() waits till the crypto driver object is idle again, this could lead to a dead lock in combination with the following requirement:
[SWS_Crypto_00026] When the synchronous job processing is used, the corresponding interface functions shall compute the result synchronously within the context of this function call.

Assume a new synchronous job is requested initiated from a task T_new with a higher priority than the task T_old that initiated the currently being processed job. T_new would not be preempted by T_old but would have to wait till T_old has finished.

Could you pl. clarify this issue.

Agreed solution:

Remove [SWS_Crypto_00120].

BW-C-Level:

Application	Specification	Bus
1	1	1

1.51 Specification Item SWS_Crypto_00122

Trace References:

none

Content:

Service name:	Crypto_CancelJobCrypto_CancelJob
Syntax:	Std_ReturnType Crypto_CancelJob(uint32 objectId, Crypto_JobInfoType* job)
Service ID[hex]:	0x0e
Sync/Async:	Synchronous
Reentrancy:	Reentrant, but not for same Crypto Driver Object

Parameters (in):	objectIdCrypto_CancelJob.objectId	Holds the identifier of the Crypto Driver Object.
Parameters (inout):	jobCrypto_CancelJob.job	Pointer to the configuration of the job. Contains structures with job 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. CRYPTO_E_JOB_CANCELED: The job has been cancelled but is still processed. No results will be returned to the application.
Description:	This interface removes the provided job from the queue and cancels the processing of the job if possible.	

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77263: [CRYPTO] API function argument naming between Csm, CryIf and Crypto

  Problem description:

  In [SWS_Csm_01051] "Csm_RandomSeed" and [SWS_CryIf_91007] "CryIf_RandomSeed" function arguments are named "seedPtr and "seedLength". In [SWS_Crypto_91013] "Crypto_RandomSeed" the same arguments are named "entropyPtr" and "entropyLength".

  Agreed solution:

  [SWS_Crypto_91013]

  rename argument "entropyPtr to "seedPtr"
  rename argument "entropyLengthto "seedLength"

  Correct argument description as well
  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
  –Last change on issue 77263 comment 6–

  BW-C-Level:

Application	Specification	Bus
1	1	1

1.52 Specification Item SWS_Crypto_00123

Trace References:

none

Content:

If default development error detection for the Crypto Driver is enabled: The function Crypto_CancelJob shall raise the error CRYPTO_E_UNINIT and return E_NOT_OK if the module is not yet initialized.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]

[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]

[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]

[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.53 Specification Item SWS_Crypto_00124

Trace References:

none

Content:

If default development error detection for the Crypto Driver is enabled: The function Crypto_CancelJob shall raise the error CRYPTO_E_PARAM_HANDLE and return E_NOT_OK if the parameter objectId is out or range.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

Problem description:

replace "default error detection" with "development error detection"
–Last change on issue 76932 comment 2–

Agreed solution:

SWS_CryIf:
replace "default error" detection with "development error detection" in requirement:
[SWS_CryIf_00016]
[SWS_CryIf_00017]
[SWS_CryIf_00027]
[SWS_CryIf_00028]
[SWS_CryIf_00029]
[SWS_CryIf_00129]
[SWS_CryIf_00130]
[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]

[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]

[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]

[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.54 Specification Item SWS_Crypto_00125

Trace References:

none

Content:

If default development error detection for the Crypto Driver is enabled: The function Crypto_CancelJob shall raise the error CRYPTO_E_PARAM_POINTER and return E_NOT_OK if the parameter job is a null pointer.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]

[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]

[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]

[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.55 Specification Item SWS_Crypto_00128

Trace References:

none

Content:

If the module is not yet initialized and if default development error detection for the Crypto Driver is enabled, the function Crypto_RandomSeed shall report CRYPTO_E_UNINIT to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]

[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]

[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]

[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.56 Specification Item SWS_Crypto_00129

Trace References:

none

Content:

If the parameter cryptoKeyId is out of range and if default development error detection for the Crypto Driver is enabled, the function Crypto_RandomSeed shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

Problem description:

replace "default error detection" with "development error detection"
–Last change on issue 76932 comment 2–

Agreed solution:

SWS_CryIf:
replace "default error" detection with "development error detection" in requirement:
[SWS_CryIf_00016]
[SWS_CryIf_00017]
[SWS_CryIf_00027]
[SWS_CryIf_00028]
[SWS_CryIf_00029]
[SWS_CryIf_00129]
[SWS_CryIf_00130]
[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]

[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]

[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]

[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.57 Specification Item SWS_Crypto_00130

Trace References:

none

Content:

If the parameter seedPtr is a null pointer and if default development error detection for the Crypto Driver is enabled, the function Crypto_RandomSeed shall report CRYPTO_E_PARAM_POINTER to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76783: Typo or copy/paste mistakes

  Problem description:

  Hello,

  I found some other mistakes in the specification documents. Most of them are typos or copy/paste mistakes. As document owner of the CryptoServiceManager, I need a confirmation from someone else, before I can implement them into the document.

  AUTOSAR_SWS_CryptoDriver:
  [SWS_Crypto_00139]: CRYPTO_E_KEY_EXTRACT_DENIED does not exist anymore. Replace error code with CRYPTO_E_KEY_READ_FAIL.
  [SWS_Crypto_91005]: Crypto_KeyValidSet() shall be named analogously to Csm_KeySetValid() and CryIf_KeySetValid(). Therefore, rename Crypto_KeyValidSet() to Crypto_KeySetValid().

[SWS_Crypto_00071]: In table: inputLengthPtr, secondaryInputLengthPtr, tertiaryInputLengthPtr are no pointer anymore. rename them to inputLength, secondaryInputLength, tertiaryInputLength

AUTOSAR_SWS_CryptoServiceManager:
[SWS_Csm_01035]: Csm_KeyCopy() shall call CryIf_KeyCopy() not CryIf_KeyElementCopy().
[SWS_Csm_01080]: Csm_AsymPrivateKeyType is not up-to-date. It should be modified like [SWS_Csm_00076] Csm_AsymPublicKeyType or [SWS_Csm_01082] Csm_SymKeyType.
SWS_Csm_00455
[SWS_Csm_00455]: tag as obsolete
[ECUC_Csm_00188]: typo: CsmMacGenerateAlgorithmFamiliy -> CsmMacGenerateAlgorithmFamily
[ECUC_Csm_00049]: CsmMacVerifyAlgorithmMode missing. (see analogues CsmMacGenerateAlgorithmMode [ECUC_Csm_00189])
[SWS_Csm_00966]: CopyPaste mistake: Delete: "Wrong return values - here are the correct ones:"
[SWS_Csm_01023]: plaintextLength description wrong. replace with "Contains the number of bytes to encrypt."
[SWS_Csm_01023]: typo "associtatedDataLengthPtr" and it is no pointer. replace with: "associatedDataLength"
[SWS_Csm_01025]: typo, replace line "job->jobPrimitiveInputOutput.outputLength = ciphertextLength," with
"job->jobPrimitiveInputOutput.outputLengthPtr = ciphertextLengthPtr,"
[SWS_Csm_01013]: typo: rename "PrimitiveInputOutput" to "jobPrimitiveInputOutput". Or is this rename intended?
Then every assignment of "jobPrimitiveInputOutput" has to be renamed to "primitiveInputOutput" like
"job->jobPrimitiveInputOutput.mode = mode," has to be modified to "job->primitiveInputOutput.mode = mode,"
[SWS_Csm_01026]: typo: replace "associtatedDataLength" with "associatedDataLength"
[SWS_Csm_01027]: missing line: "job->jobPrimitiveInputOutput.verifyPtr = verifyPtr."
[SWS_Csm_00992]: copypaste mistake: replace "mode: The Crypto_JobInfoType job with the corresponding jobId shall be modified in the following way:" with ""mode: Indicates which operation mode(s) to perfom."
[SWS_Csm_00992]: copypaste mistake: replace "resultLengthPtr: Contains the number of bytes of the associated data." with ""resultLengthPtr: Holds a pointer to the memory location in which the output length in bytes of the signature is stored. On calling this function, this parameter shall contain the size of the buffer provided

by resultPtr. When the request has finished, the actual length of the returned value shall be stored."
[SWS_Csm_01543]: description wrong. replace with "Generate a random number and stores it in the memory location pointed by the result pointer."
[SWS_Csm_00168]: description wrong, there is no IV. replace with "This function is deprecated. Sets the key for symmetrical encryption."
[SWS_Csm_01031]: description wrong, it is not decrement. "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterRead Service"

A proposed solution is added, too.

Agreed solution:

AUTOSAR_SWS_CryptoDriver:
[SWS_Crypto_00139]: Replace CRYPTO_E_KEY_EXTRACT_DENIED with CRYPTO_E_KEY_READ_FAIL.
[SWS_CryIf_91015]: Remove CRYPTO_E_KEY_EXTRACT_DENIED
[SWS_Crypto_91005]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add Crypto_KeySetValid as API (Description according to SWS_Crypto_91005)
[SWS_Crypto_00082]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add E_UNINIT DET check SWS for Crypto_KeySetValid (Text according to SWS_Crypto_00082)
[SWS_Crypto_00083]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add E_PARAM_HANDLE DET check SWS for Crypto_KeySetValid (Text according to SWS_Crypto_00083)
last sentence in 8.2.4.1.2: Rename Crypto_KeyValidSet to Crypto_KeySetValid

[SWS_Crypto_00071]: rename inputLengthPtr, secondaryInputLengthPtr, tertiaryInputLengthPtr to inputLength, secondaryInputLength, tertiaryInputLength

AUTOSAR_SWS_CryptoServiceManager:
[SWS_Csm_01035]: CryIf_KeyElementCopy() shall be replaced with CryIf_KeyCopy().

[SWS_Csm_01080]: replace with (see [SWS_Csm_00076]):
Name: Csm_AsymPrivateKeyType
Kind: Structure
Elements:
length: uint32: This element contains the length in bytes of the key stored in element 'data'
data: Csm_AsymPrivateKeyArrayType: This element contains the key data or a key handle.
Description: Structure for the private asymmetrical key.

Variation: –

[SWS_Csm_00455]: tag as obsolete

[ECUC_Csm_00188]: typo: CsmMacGenerateAlgorithmFamiliy -> CsmMacGenerateAlgorithmFamily
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmMode (see analogues CsmMacGenerateAlgorithmMode [ECUC_Csm_00189])
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmModeCustom (see analogues CsmMacGenerateAlgorithmModeCustom [ECUC_Csm_00189])
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmKeyLength (see analogues CsmMacGenerateAlgorithmKeyLength [ECUC_Csm_00189])
[SWS_Csm_00966]: Delete: "Wrong return values - here are the correct ones:"
[SWS_Csm_01023]: Replace description with: "Contains the number of bytes to encrypt."
[SWS_Csm_01023]: Replace "associatedDataLengthPtr" with "associatedDataLength"
[SWS_Csm_01025]: Replace line "job->jobPrimitiveInputOutput.outputLength = ciphertextLength," with
"job->jobPrimitiveInputOutput.outputLengthPtr = ciphertextLengthPtr,"
[SWS_Csm_01013]: rename "PrimitiveInputOutput" to "jobPrimitiveInputOutput". rename "state" to "jobState".
[SWS_Csm_01026]: replace "associtatedDataLength" with "associatedDataLength"
[SWS_Csm_01027]: add line: "job->jobPrimitiveInputOutput.verifyPtr = verifyPtr."
[SWS_Csm_00992]: replace "mode: The Crypto_JobInfoType job with the corresponding jobId shall be modified in the following way:" with ""mode: Indicates which operation mode(s) to perform."
[SWS_Csm_00992]: replace "resultLengthPtr: Contains the number of bytes of the associated data." with ""resultLengthPtr: Holds a pointer to the memory location in which the output length in bytes of the signature is stored. On calling this function, this parameter shall contain the size of the buffer provided by resultPtr. When the request has finished, the actual length of the returned value shall be stored."
[SWS_Csm_01543]: replace description with "Generate a random number and stores it in the memory location pointed by the result pointer."
[SWS_Csm_00168]: replace description with "This function is deprecated. Sets the key for symmetrical encryption."
[SWS_Csm_01031]: replace "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterDecrement Service" with "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterRead Service"
–Last change on issue 76783 comment 29–

BW-C-Level:

Application	Specification	Bus
4	3	1

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]
  [SWS_CryIf_00112]
  [SWS_CryIf_00116]
  [SWS_CryIf_00117]
  [SWS_CryIf_00118]
  [SWS_CryIf_00068]
  [SWS_CryIf_00069]
  [SWS_CryIf_00070]
  [SWS_CryIf_00071]
  [SWS_CryIf_00073]
  [SWS_CryIf_00074]

[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]

[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]

[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

- RfC #77937: [CRYPTO] Parameters inconsistences in SWS Crypto

Problem description:

Hello,
Please verify the following inconsistencies between naming inside AUTOSAR_SWS_CryptoDriver:

1.
[SWS_Crypto_91013] the parameter name is entropyLength
[SWS_Crypto_00131] the parameter name is seedLength
Is parameter name seedLength or entropyLength?

2.
[SWS_Crypto_91010] the parameter name is partnerPublicValueLength
[SWS_Crypto_00115] the parameter name is partnerPubValueLength.
Is parameter name partnerPublicValueLength or partnerPubValueLength?

3.
[SWS_Crypto_00171] the parameter name is verifyCryptoKeyId
[SWS_Crypto_00174] the parameter name is validateCryptoKeyId
Is parameter name verifyCryptoKeyId or validateCryptoKeyId?

Thank you,
Alexandra

Agreed solution:

[SWS_Crypto_00130] Replace seedPtr with entropyPtr
[SWS_Crypto_00131] Replace seedLength with entropyLength.
[SWS_Crypto_00115] Replace partnerPubValueLength with partnerPublicValueLength.
[SWS_Crypto_00174] Replace validateCryptoKeyId with verifyCryptoKeyId
–Last change on issue 77937 comment 3–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.58 Specification Item SWS_Crypto_00131

Trace References:

none

Content:

If seedLength is zero and if default development error detection for the Crypto Driver is enabled, the function Crypto_RandomSeed shall report CRYPTO_E_PARAM_VALUE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76783: Typo or copy/paste mistakes

  Problem description:

  Hello,

  I found some other mistakes in the specification documents. Most of them are typos or copy/paste mistakes. As document owner of the CryptoServiceManager, I need a confirmation from someone else, before I can implement them into the document.

  AUTOSAR_SWS_CryptoDriver:
  [SWS_Crypto_00139]: CRYPTO_E_KEY_EXTRACT_DENIED does not exist anymore. Replace error code with CRYPTO_E_KEY_READ_FAIL.
  [SWS_Crypto_91005]: Crypto_KeyValidSet() shall be named analogously to Csm_KeySetValid() and CryIf_KeySetValid(). Therefore, rename Crypto_KeyValidSet() to Crypto_KeySetValid().
  [SWS_Crypto_00071]: In table: inputLengthPtr, secondaryInputLengthPtr, tertiaryInputLengthPtr are no pointer anymore. rename them to inputLength,

secondaryInputLength, tertiaryInputLength

AUTOSAR_SWS_CryptoServiceManager:
[SWS_Csm_01035]: Csm_KeyCopy() shall call CryIf_KeyCopy() not CryIf_KeyElementCopy().
[SWS_Csm_01080]: Csm_AsymPrivateKeyType is not up-to-date. It should be modified like [SWS_Csm_00076] Csm_AsymPublicKeyType or [SWS_Csm_01082] Csm_SymKeyType.
SWS_Csm_00455
[SWS_Csm_00455]: tag as obsolete
[ECUC_Csm_00188]: typo: CsmMacGenerateAlgorithmFamiliy -> CsmMacGenerateAlgorithmFamily
[ECUC_Csm_00049]: CsmMacVerifyAlgorithmMode missing. (see analogues CsmMacGenerateAlgorithmMode [ECUC_Csm_00189])
[SWS_Csm_00966]: CopyPaste mistake: Delete: "Wrong return values - here are the correct ones:"
[SWS_Csm_01023]: plaintextLength description wrong. replace with "Contains the number of bytes to encrypt."
[SWS_Csm_01023]: typo "associtatedDataLengthPtr" and it is no pointer. replace with: "associatedDataLength"
[SWS_Csm_01025]: typo, replace line "job->jobPrimitiveInputOutput.outputLength = ciphertextLength," with
"job->jobPrimitiveInputOutput.outputLengthPtr = ciphertextLengthPtr,"
[SWS_Csm_01013]: typo: rename "PrimitiveInputOutput" to "jobPrimitiveInputOutput". Or is this rename intended?
Then every assignment of "jobPrimitiveInputOutput" has to be renamed to "primitiveInputOutput" like
"job->jobPrimitiveInputOutput.mode = mode," has to be modified to "job->primitiveInputOutput.mode = mode,"
[SWS_Csm_01026]: typo: replace "associtatedDataLength" with "associatedDataLength"
[SWS_Csm_01027]: missing line: "job->jobPrimitiveInputOutput.verifyPtr = verifyPtr."
[SWS_Csm_00992]: copypaste mistake: replace "mode: The Crypto_JobInfoType job with the corresponding jobId shall be modified in the following way:" with ""mode: Indicates which operation mode(s) to perfom."
[SWS_Csm_00992]: copypaste mistake: replace "resultLengthPtr: Contains the number of bytes of the associated data." with ""resultLengthPtr: Holds a pointer to the memory location in which the output length in bytes of the signature is stored. On calling this function, this parameter shall contain the size of the buffer provided by resultPtr. When the request has finished, the actual length of the returned value shall be stored."

[SWS_Csm_01543]: description wrong. replace with "Generate a random number and stores it in the memory location pointed by the result pointer."
[SWS_Csm_00168]: description wrong, there is no IV. replace with "This function is deprecated. Sets the key for symmetrical encryption."
[SWS_Csm_01031]: description wrong, it is not decrement. "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterRead Service"

A proposed solution is added, too.

Agreed solution:

AUTOSAR_SWS_CryptoDriver:
[SWS_Crypto_00139]: Replace CRYPTO_E_KEY_EXTRACT_DENIED with CRYPTO_E_KEY_READ_FAIL.
[SWS_CryIf_91015]: Remove CRYPTO_E_KEY_EXTRACT_DENIED
[SWS_Crypto_91005]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add Crypto_KeySetValid as API (Description according to SWS_Crypto_91005)
[SWS_Crypto_00082]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add E_UNINIT DET check SWS for Crypto_KeySetValid (Text according to SWS_Crypto_00082)
[SWS_Crypto_00083]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add E_PARAM_HANDLE DET check SWS for Crypto_KeySetValid (Text according to SWS_Crypto_00083)
last sentence in 8.2.4.1.2: Rename Crypto_KeyValidSet to Crypto_KeySetValid

[SWS_Crypto_00071]: rename inputLengthPtr, secondaryInputLengthPtr, tertiaryInputLengthPtr to inputLength, secondaryInputLength, tertiaryInputLength

AUTOSAR_SWS_CryptoServiceManager:
[SWS_Csm_01035]: CryIf_KeyElementCopy() shall be replaced with CryIf_KeyCopy().

[SWS_Csm_01080]: replace with (see [SWS_Csm_00076]):
Name: Csm_AsymPrivateKeyType
Kind: Structure
Elements:
length: uint32: This element contains the length in bytes of the key stored in element ’data’
data: Csm_AsymPrivateKeyArrayType: This element contains the key data or a key handle.
Description: Structure for the private asymmetrical key.
Variation: –

[SWS_Csm_00455]: tag as obsolete

[ECUC_Csm_00188]: typo: CsmMacGenerateAlgorithmFamiliy -> CsmMacGenerateAlgorithmFamily
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmMode (see analogues CsmMacGenerateAlgorithmMode [ECUC_Csm_00189])
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmModeCustom (see analogues CsmMacGenerateAlgorithmModeCustom [ECUC_Csm_00189])
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmKeyLength (see analogues CsmMacGenerateAlgorithmKeyLength [ECUC_Csm_00189])
[SWS_Csm_00966]: Delete: "Wrong return values - here are the correct ones:"
[SWS_Csm_01023]: Replace description with: "Contains the number of bytes to encrypt."
[SWS_Csm_01023]: Replace "associatedDataLengthPtr" with "associatedDataLength"
[SWS_Csm_01025]: Replace line "job->jobPrimitiveInputOutput.outputLength = ciphertextLength," with
"job->jobPrimitiveInputOutput.outputLengthPtr = ciphertextLengthPtr,"
[SWS_Csm_01013]: rename "PrimitiveInputOutput" to "jobPrimitiveInputOutput". rename "state" to "jobState".
[SWS_Csm_01026]: replace "associtatedDataLength" with "associatedDataLength"
[SWS_Csm_01027]: add line: "job->jobPrimitiveInputOutput.verifyPtr = verifyPtr."
[SWS_Csm_00992]: replace "mode: The Crypto_JobInfoType job with the corresponding jobId shall be modified in the following way:" with ""mode: Indicates which operation mode(s) to perform."
[SWS_Csm_00992]: replace "resultLengthPtr: Contains the number of bytes of the associated data." with ""resultLengthPtr: Holds a pointer to the memory location in which the output length in bytes of the signature is stored. On calling this function, this parameter shall contain the size of the buffer provided by resultPtr. When the request has finished, the actual length of the returned value shall be stored."
[SWS_Csm_01543]: replace description with "Generate a random number and stores it in the memory location pointed by the result pointer."
[SWS_Csm_00168]: replace description with "This function is deprecated. Sets the key for symmetrical encryption."
[SWS_Csm_01031]: replace "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterDecrement Service" with "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterRead Service"
–Last change on issue 76783 comment 29–

BW-C-Level:

Application	Specification	Bus
4	3	1

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]
  [SWS_CryIf_00112]
  [SWS_CryIf_00116]
  [SWS_CryIf_00117]
  [SWS_CryIf_00118]
  [SWS_CryIf_00068]
  [SWS_CryIf_00069]
  [SWS_CryIf_00070]
  [SWS_CryIf_00071]
  [SWS_CryIf_00073]
  [SWS_CryIf_00074]
  [SWS_CryIf_00076]
  [SWS_CryIf_00077]
  [SWS_CryIf_00122]

[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]

[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]

[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

- RfC #77937: [CRYPTO] Parameters inconsistences in SWS Crypto

  Problem description:

  Hello,
  Please verify the following inconsistencies between naming inside AUTOSAR_SWS_CryptoDriver:

  1.
  [SWS_Crypto_91013] the parameter name is entropyLength
  [SWS_Crypto_00131] the parameter name is seedLength
  Is parameter name seedLength or entropyLength?

  2.
  [SWS_Crypto_91010] the parameter name is partnerPublicValueLength
  [SWS_Crypto_00115] the parameter name is partnerPubValueLength.
  Is parameter name partnerPublicValueLength or partnerPubValueLength?

  3.
  [SWS_Crypto_00171] the parameter name is verifyCryptoKeyId
  [SWS_Crypto_00174] the parameter name is validateCryptoKeyId
  Is parameter name verifyCryptoKeyId or validateCryptoKeyId?

  Thank you,
  Alexandra

  Agreed solution:

  [SWS_Crypto_00130] Replace seedPtr with entropyPtr
  [SWS_Crypto_00131] Replace seedLength with entropyLength.

[SWS_Crypto_00115] Replace partnerPubValueLength with partnerPublicValueLength.
[SWS_Crypto_00174] Replace validateCryptoKeyId with verifyCryptoKeyId
–Last change on issue 77937 comment 3–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.59 Specification Item SWS_Crypto_00136

Trace References:

none

Content:

If the buffer job->jobPrimitiveInput.outputPtr or job->jobPrimitiveInput.secondaryOutputPtr is too small to store the result of the request, CRYPTO_E_SMALL_BUFFER shall be returned and if default error detection is enabled, the function shall additionally report the runtime error CRYPTO_E_RE_SMALL_BUFFERshall be reported to the DET.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76636: Rollout of ’Runtime errors’ for entire crypto stack documents

  Problem description:

  Crypto Stack documents are not in line with the RfC # 59085.

  In SWS_secureOnboardCommunication
  Example1: SECOC_E_CRYPTO_FAILURE in the is a development error, but should be a runtime error.

  In SWS_CryptoServiceManager
  Example2: CSM_E_SERVICE_NOT_STARTED is not referenced.

  Example3: CSM_E_PARAM_HANDLE is not referenced in chapter 7.3. It is not clear development error or runtime error.
  –Last change on issue 76636 comment 33–

  Agreed solution:

CryptoInterface:
https://bugzilla.autosar.org/attachment.cgi?id=4587

CryptoServiceManager:
https://bugzilla.autosar.org/attachment.cgi?id=4614

CryptoDriver:
https://bugzilla.autosar.org/attachment.cgi?id=4613

SecureOnboardCommunication:
https://bugzilla.autosar.org/attachment.cgi?id=4598
–Last change on issue 76636 comment 41–

BW-C-Level:

Application	Specification	Bus
1	4	1

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]

[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]

[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]

[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.60 Specification Item SWS_Crypto_00137

Trace References:

none

Content:

If the increment secure counter service is chosen and the corresponding counter is overflowed and default development error detection for the Crypto Driver is enabled, the function Crypto_ProcessJob shall report CRYPTO_E_PARAM_HANDLE to the DET and return CRYPTO_E_COUNTER_OVERFLOW.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]
  [SWS_CryIf_00112]
  [SWS_CryIf_00116]
  [SWS_CryIf_00117]
  [SWS_CryIf_00118]
  [SWS_CryIf_00068]
  [SWS_CryIf_00069]
  [SWS_CryIf_00070]
  [SWS_CryIf_00071]
  [SWS_CryIf_00073]
  [SWS_CryIf_00074]
  [SWS_CryIf_00076]

[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]

[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]

[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.61 Specification Item SWS_Crypto_00139

Trace References:

none

Content:

If the function Crypto_KeyElementGet returns CRYPTO_E_KEY_EXTRACT_DENIED and default error detection is enabledREAD_FAIL, the function shall additionally report the runtime error CRYPTO_E_RE_KEY_EXTRACT_DENIED to the DETREAD_FAIL.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76636: Rollout of ’Runtime errors’ for entire crypto stack documents

  Problem description:

  Crypto Stack documents are not in line with the RfC # 59085.

  In SWS_secureOnboardCommunication
  Example1: SECOC_E_CRYPTO_FAILURE in the is a development error, but should be a runtime error.

  In SWS_CryptoServiceManager
  Example2: CSM_E_SERVICE_NOT_STARTED is not referenced.

Example3: CSM_E_PARAM_HANDLE is not referenced in chapter 7.3. It is not clear development error or runtime error.
–Last change on issue 76636 comment 33–

Agreed solution:

CryptoInterface:
https://bugzilla.autosar.org/attachment.cgi?id=4587

CryptoServiceManager:
https://bugzilla.autosar.org/attachment.cgi?id=4614

CryptoDriver:
https://bugzilla.autosar.org/attachment.cgi?id=4613

SecureOnboardCommunication:
https://bugzilla.autosar.org/attachment.cgi?id=4598
–Last change on issue 76636 comment 41–

BW-C-Level:

Application	Specification	Bus
1	4	1

- RfC #76783: Typo or copy/paste mistakes

Problem description:

Hello,

I found some other mistakes in the specification documents. Most of them are typos or copy/paste mistakes. As document owner of the CryptoServiceManager, I need a confirmation from someone else, before I can implement them into the document.

AUTOSAR_SWS_CryptoDriver:
[SWS_Crypto_00139]: CRYPTO_E_KEY_EXTRACT_DENIED does not exist anymore. Replace error code with CRYPTO_E_KEY_READ_FAIL.
[SWS_Crypto_91005]: Crypto_KeyValidSet() shall be named analogously to Csm_KeySetValid() and CryIf_KeySetValid(). Therefore, rename Crypto_KeyValidSet() to Crypto_KeySetValid().
[SWS_Crypto_00071]: In table: inputLengthPtr, secondaryInputLengthPtr, tertiaryInputLengthPtr are no pointer anymore. rename them to inputLength, secondaryInputLength, tertiaryInputLength

AUTOSAR_SWS_CryptoServiceManager:
[SWS_Csm_01035]: Csm_KeyCopy() shall call CryIf_KeyCopy() not CryIf_KeyElementCopy().
[SWS_Csm_01080]: Csm_AsymPrivateKeyType is not up-to-date. It should be modified like [SWS_Csm_00076] Csm_AsymPublicKeyType or [SWS_Csm_01082] Csm_SymKeyType.
SWS_Csm_00455
[SWS_Csm_00455]: tag as obsolete
[ECUC_Csm_00188]: typo: CsmMacGenerateAlgorithmFamiliy -> CsmMacGenerateAlgorithmFamily
[ECUC_Csm_00049]: CsmMacVerifyAlgorithmMode missing. (see analogues CsmMacGenerateAlgorithmMode [ECUC_Csm_00189])
[SWS_Csm_00966]: CopyPaste mistake: Delete: "Wrong return values - here are the correct ones:"
[SWS_Csm_01023]: plaintextLength description wrong. replace with "Contains the number of bytes to encrypt."
[SWS_Csm_01023]: typo "associtatedDataLengthPtr" and it is no pointer. replace with: "associatedDataLength"
[SWS_Csm_01025]: typo, replace line "job->jobPrimitiveInputOutput.outputLength = ciphertextLength," with
"job->jobPrimitiveInputOutput.outputLengthPtr = ciphertextLengthPtr,"
[SWS_Csm_01013]: typo: rename "PrimitiveInputOutput" to "jobPrimitiveInputOutput". Or is this rename intended?
Then every assignment of "jobPrimitiveInputOutput" has to be renamed to "primitiveInputOutput" like
"job->jobPrimitiveInputOutput.mode = mode," has to be modified to "job->primitiveInputOutput.mode = mode,"
[SWS_Csm_01026]: typo: replace "associtatedDataLength" with "associatedDataLength"
[SWS_Csm_01027]: missing line: "job->jobPrimitiveInputOutput.verifyPtr = verifyPtr."
[SWS_Csm_00992]: copypaste mistake: replace "mode: The Crypto_JobInfoType job with the corresponding jobId shall be modified in the following way:" with ""mode: Indicates which operation mode(s) to perfom."
[SWS_Csm_00992]: copypaste mistake: replace "resultLengthPtr: Contains the number of bytes of the associated data." with ""resultLengthPtr: Holds a pointer to the memory location in which the output length in bytes of the signature is stored. On calling this function, this parameter shall contain the size of the buffer provided by resultPtr. When the request has finished, the actual length of the returned value shall be stored."
[SWS_Csm_01543]: description wrong. replace with "Generate a random number and stores it in the memory location pointed by the result pointer."
[SWS_Csm_00168]: description wrong, there is no IV. replace with "This function is

deprecated. Sets the key for symmetrical encryption."
[SWS_Csm_01031]: description wrong, it is not decrement. "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterRead Service"

A proposed solution is added, too.

Agreed solution:

AUTOSAR_SWS_CryptoDriver:
[SWS_Crypto_00139]: Replace CRYPTO_E_KEY_EXTRACT_DENIED with CRYPTO_E_KEY_READ_FAIL.
[SWS_CryIf_91015]: Remove CRYPTO_E_KEY_EXTRACT_DENIED
[SWS_Crypto_91005]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add Crypto_KeySetValid as API (Description according to SWS_Crypto_91005)
[SWS_Crypto_00082]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add E_UNINIT DET check SWS for Crypto_KeySetValid (Text according to SWS_Crypto_00082)
[SWS_Crypto_00083]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add E_PARAM_HANDLE DET check SWS for Crypto_KeySetValid (Text according to SWS_Crypto_00083)
last sentence in 8.2.4.1.2: Rename Crypto_KeyValidSet to Crypto_KeySetValid

[SWS_Crypto_00071]: rename inputLengthPtr, secondaryInputLengthPtr, tertiaryInputLengthPtr to inputLength, secondaryInputLength, tertiaryInputLength

AUTOSAR_SWS_CryptoServiceManager:
[SWS_Csm_01035]: CryIf_KeyElementCopy() shall be replaced with CryIf_KeyCopy().

[SWS_Csm_01080]: replace with (see [SWS_Csm_00076]):
Name: Csm_AsymPrivateKeyType
Kind: Structure
Elements:
length: uint32: This element contains the length in bytes of the key stored in element 'data'
data: Csm_AsymPrivateKeyArrayType: This element contains the key data or a key handle.
Description: Structure for the private asymmetrical key.
Variation: –

[SWS_Csm_00455]: tag as obsolete

[ECUC_Csm_00188]: typo: CsmMacGenerateAlgorithmFamiliy -> CsmMac-

GenerateAlgorithmFamily
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmMode (see analogues CsmMacGenerateAlgorithmMode [ECUC_Csm_00189])
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmModeCustom (see analogues CsmMacGenerateAlgorithmModeCustom [ECUC_Csm_00189])
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmKeyLength (see analogues CsmMacGenerateAlgorithmKeyLength [ECUC_Csm_00189])
[SWS_Csm_00966]: Delete: "Wrong return values - here are the correct ones:"
[SWS_Csm_01023]: Replace description with: "Contains the number of bytes to encrypt."
[SWS_Csm_01023]: Replace "associatedDataLengthPtr" with "associatedDataLength"
[SWS_Csm_01025]: Replace line "job->jobPrimitiveInputOutput.outputLength = ciphertextLength," with
"job->jobPrimitiveInputOutput.outputLengthPtr = ciphertextLengthPtr,"
[SWS_Csm_01013]: rename "PrimitiveInputOutput" to "jobPrimitiveInputOutput". rename "state" to "jobState".
[SWS_Csm_01026]: replace "associtatedDataLength" with "associatedDataLength"
[SWS_Csm_01027]: add line: "job->jobPrimitiveInputOutput.verifyPtr = verifyPtr."
[SWS_Csm_00992]: replace "mode: The Crypto_JobInfoType job with the corresponding jobId shall be modified in the following way:" with ""mode: Indicates which operation mode(s) to perform."
[SWS_Csm_00992]: replace "resultLengthPtr: Contains the number of bytes of the associated data." with ""resultLengthPtr: Holds a pointer to the memory location in which the output length in bytes of the signature is stored. On calling this function, this parameter shall contain the size of the buffer provided by resultPtr. When the request has finished, the actual length of the returned value shall be stored."
[SWS_Csm_01543]: replace description with "Generate a random number and stores it in the memory location pointed by the result pointer."
[SWS_Csm_00168]: replace description with "This function is deprecated. Sets the key for symmetrical encryption."
[SWS_Csm_01031]: replace "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterDecrement Service" with "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterRead Service"
–Last change on issue 76783 comment 29–

BW-C-Level:

Application	Specification	Bus
4	3	1

1.62 Specification Item SWS_Crypto_00140

Trace References:

none

Content:

If the function Crypto_KeyElementGet returns CRYPTO_E_KEY_NOT_AVAILABLEand default error detection is enabled, the function shall additionally report the runtime error CRYPTO_E_RE_KEY_NOT_AVAILABLEto the DET.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76636: Rollout of 'Runtime errors' for entire crypto stack documents

  Problem description:

  Crypto Stack documents are not in line with the RfC # 59085.

  In SWS_secureOnboardCommunication
  Example1: SECOC_E_CRYPTO_FAILURE in the is a development error, but should be a runtime error.

  In SWS_CryptoServiceManager
  Example2: CSM_E_SERVICE_NOT_STARTED is not referenced.

  Example3: CSM_E_PARAM_HANDLE is not referenced in chapter 7.3. It is not clear development error or runtime error.
  –Last change on issue 76636 comment 33–

  Agreed solution:

  CryptoInterface:
  https://bugzilla.autosar.org/attachment.cgi?id=4587

  CryptoServiceManager:
  https://bugzilla.autosar.org/attachment.cgi?id=4614

  CryptoDriver:
  https://bugzilla.autosar.org/attachment.cgi?id=4613

  SecureOnboardCommunication:
  https://bugzilla.autosar.org/attachment.cgi?id=4598
  –Last change on issue 76636 comment 41–

BW-C-Level:

Application	Specification	Bus
1	4	1

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]
  [SWS_CryIf_00112]
  [SWS_CryIf_00116]
  [SWS_CryIf_00117]
  [SWS_CryIf_00118]
  [SWS_CryIf_00068]
  [SWS_CryIf_00069]
  [SWS_CryIf_00070]
  [SWS_CryIf_00071]
  [SWS_CryIf_00073]

[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]

[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]

[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.63 Specification Item SWS_Crypto_00141

Trace References:

none

Content:

If the random generator service is chosen and the corresponding entropy, the function shall return CRYPTO_E_ENTROPY_EXHAUSTED. If the default error detection for the Crypto Driver is enabled, the The function Crypto_ProcessJob shall additionally report the runtime error CRYPTO_E_RE_ENTROPY_EXHAUSTEDto the DET.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76636: Rollout of 'Runtime errors' for entire crypto stack documents

  Problem description:

  Crypto Stack documents are not in line with the RfC # 59085.

  In SWS_secureOnboardCommunication
  Example1: SECOC_E_CRYPTO_FAILURE in the is a development error, but should be a runtime error.

In SWS_CryptoServiceManager
Example2: CSM_E_SERVICE_NOT_STARTED is not referenced.

Example3: CSM_E_PARAM_HANDLE is not referenced in chapter 7.3. It is not clear development error or runtime error.
–Last change on issue 76636 comment 33–

Agreed solution:

CryptoInterface:
https://bugzilla.autosar.org/attachment.cgi?id=4587

CryptoServiceManager:
https://bugzilla.autosar.org/attachment.cgi?id=4614

CryptoDriver:
https://bugzilla.autosar.org/attachment.cgi?id=4613

SecureOnboardCommunication:
https://bugzilla.autosar.org/attachment.cgi?id=4598
–Last change on issue 76636 comment 41–

BW-C-Level:

Application	Specification	Bus
1	4	1

- RfC #76932: default error detection -> development error detection

Problem description:

replace "default error detection" with "development error detection"
–Last change on issue 76932 comment 2–

Agreed solution:

SWS_CryIf:
replace "default error" detection with "development error detection" in requirement:
[SWS_CryIf_00016]
[SWS_CryIf_00017]
[SWS_CryIf_00027]
[SWS_CryIf_00028]
[SWS_CryIf_00029]
[SWS_CryIf_00129]
[SWS_CryIf_00130]
[SWS_CryIf_00131]
[SWS_CryIf_00049]

[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]

[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]

[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.64 Specification Item SWS_Crypto_00142

Trace References:

none

Content:

If a length pointer is required as an argument, but the value, which is pointed by the length pointer is zero, and if default development error detection for the Crypto Driver is enabled, the Crypto_ProcessJob() function report CRYPTO_E_PARAM_VALUE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]

[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]

[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]

[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.65 Specification Item SWS_Crypto_00143

Trace References:

none

Content:

If no errors are detected by Crypto Driver, the service Crypto_CancelJob() shall remove the job from the queue. If the job is currently processed it shall be cancelled. When cancellation of current processing is not possible due to limitations, the result shall be discarded and the callback notification shall be suppressed.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77374: Postponed Crypto_CancelJob()

  Problem description:

  If a job cannot be canceled by Crypto_CancelJob() immediately, it is not clear how to proceed. The requirements say:

  [SWS_Crypto_00143] If no errors are detected by Crypto Driver, the service Crypto_CancelJob() shall remove the job from the queue. If the job is currently processed it shall be cancelled. When cancellation of current processing is not possible due to limitations, the result shall be discarded and the callback notification shall be suppressed.

  [SWS_Crypto_00144] If a job is canceled, it shall return CRYPTO_E_JOB_CANCELED either with the callback, when the job is an asynchronous job or as the return value of the function Crypto_CancelJob(), in case the job is synchronous.

  The following questions arise:
  (i) Is it meant in [SWS_Crypto_00143] that (only) the notification of the finished job shall be suppressed?
  (ii) [SWS_Crypto_00144]: There is no return value CRYPTO_E_JOB_CANCELED of Crypto_CancelJob(). So what should be the return value?
  (iii) What does Crypto_CancelJob() return when the cancellation is not possible and it has to be postponed till the job has finished? Crypto_CancelJob() cannot wait till the job has finished.

  Could you pl. clarify these questions?

  Agreed solution:

  Attached to ticket
  –Last change on issue 77374 comment 7–

  BW-C-Level:

Application	Specification	Bus
1	1	1

1.66 Specification Item SWS_Crypto_00144

Trace References:

none

Content:

If a job is canceled, it shall return CRYPTO_E_JOB_CANCELED either with the callback, when the job is an asynchronous job or as the return value of the function Crypto_CancelProcessJob(), in case the job is synchronous.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77374: Postponed Crypto_CancelJob()

  Problem description:

  If a job cannot be canceled by Crypto_CancelJob() immediately, it is not clear how to proceed. The requirements say:

  [SWS_Crypto_00143] If no errors are detected by Crypto Driver, the service Crypto_CancelJob() shall remove the job from the queue. If the job is currently processed it shall be cancelled. When cancellation of current processing is not possible due to limitations, the result shall be discarded and the callback notification shall be suppressed.

  [SWS_Crypto_00144] If a job is canceled, it shall return CRYPTO_E_JOB_CANCELED either with the callback, when the job is an asynchronous job or as the return value of the function Crypto_CancelJob(), in case the job is synchronous.

  The following questions arise:
  (i) Is it meant in [SWS_Crypto_00143] that (only) the notification of the finished job shall be suppressed?
  (ii) [SWS_Crypto_00144]: There is no return value CRYPTO_E_JOB_CANCELED of Crypto_CancelJob(). So what should be the return value?
  (iii) What does Crypto_CancelJob() return when the cancellation is not possible and it has to be postponed till the job has finished? Crypto_CancelJob() cannot wait till the job has finished.

  Could you pl. clarify these questions?

  Agreed solution:

  Attached to ticket
  –Last change on issue 77374 comment 7–

  BW-C-Level:

Application	Specification	Bus
1	1	1

1.67 Specification Item SWS_Crypto_00149

Trace References:

none

Content:

If the Crypto Driver is not yet initialized and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyElementCopy shall report CRYPTO_E_UNINIT to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]

[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]

[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]

[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.68 Specification Item SWS_Crypto_00150

Trace References:

none

Content:

If cryptoKeyId is out of range and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyElementCopy shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

Problem description:

replace "default error detection" with "development error detection"
–Last change on issue 76932 comment 2–

Agreed solution:

SWS_CryIf:
replace "default error" detection with "development error detection" in requirement:
[SWS_CryIf_00016]
[SWS_CryIf_00017]
[SWS_CryIf_00027]
[SWS_CryIf_00028]
[SWS_CryIf_00029]
[SWS_CryIf_00129]
[SWS_CryIf_00130]
[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]

[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]

[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]

[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.69 Specification Item SWS_Crypto_00151

Trace References:

none

Content:

If targetCryptoKeyId is out of range and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyElementCopy shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]

[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]

[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]

[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.70 Specification Item SWS_Crypto_00152

Trace References:

none

Content:

If parameter keyElementId is out of range and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyElementCopy shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]

[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]

[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]

[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.71 Specification Item SWS_Crypto_00153

Trace References:

none

Content:

If parameter targetKeyElementId is out of range and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyElementCopy shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

Problem description:

replace "default error detection" with "development error detection"
–Last change on issue 76932 comment 2–

Agreed solution:

SWS_CryIf:
replace "default error" detection with "development error detection" in requirement:
[SWS_CryIf_00016]
[SWS_CryIf_00017]
[SWS_CryIf_00027]
[SWS_CryIf_00028]
[SWS_CryIf_00029]
[SWS_CryIf_00129]
[SWS_CryIf_00130]
[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]

[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]

[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]

[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.72 Specification Item SWS_Crypto_00156

Trace References:

none

Content:

If the Crypto Driver is not yet initialized and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyCopy shall report CRYPTO_E_UNINIT to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]

[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]

[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]

[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.73 Specification Item SWS_Crypto_00157

Trace References:

none

Content:

If cryptoKeyId is out of range and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyCopy shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]

[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]

[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]

[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.74 Specification Item SWS_Crypto_00158

Trace References:

none

Content:

If targetCryptoKeyId is out of range and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyCopy shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

Problem description:

replace "default error detection" with "development error detection"
–Last change on issue 76932 comment 2–

Agreed solution:

SWS_CryIf:
replace "default error" detection with "development error detection" in requirement:
[SWS_CryIf_00016]
[SWS_CryIf_00017]
[SWS_CryIf_00027]
[SWS_CryIf_00028]
[SWS_CryIf_00029]
[SWS_CryIf_00129]
[SWS_CryIf_00130]
[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]

[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]

[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]

[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.75 Specification Item SWS_Crypto_00161

Trace References:

none

Content:

If the Crypto Driver is not yet initialized and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyElementIdsGet shall report CRYPTO_E_UNINIT to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]

[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]

[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]

[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.76 Specification Item SWS_Crypto_00162

Trace References:

none

Content:

If cryptoKeyId is out of range and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyElementIdsGet shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]

[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]

[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]

[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	**Specification**	**Bus**
1	1	1

1.77 Specification Item SWS_Crypto_00163

Trace References:

none

Content:

If the value, which is pointed by keyElementIdsLengthPtr is smaller than the number of key elements in the key and if default development error detection for the Crypto Driver is enabled, the function Crypto_KeyElementIdsGet shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

Problem description:

replace "default error detection" with "development error detection"
–Last change on issue 76932 comment 2–

Agreed solution:

SWS_CryIf:
replace "default error" detection with "development error detection" in requirement:
[SWS_CryIf_00016]
[SWS_CryIf_00017]
[SWS_CryIf_00027]
[SWS_CryIf_00028]
[SWS_CryIf_00029]
[SWS_CryIf_00129]
[SWS_CryIf_00130]
[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]

[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]

[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]

[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.78 Specification Item SWS_Crypto_00164

Trace References:

none

Content:

If the buffer keyElementIdsPtr is too small to store the result of the request, CRYPTO_E_SMALL_BUFFER shall be returned and if ~~default~~ development error detection is enabled, CRYPTO_E_SMALL_BUFFER shall be reported to the DET.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]

[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]

[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]

[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.79 Specification Item SWS_Crypto_00168

Trace References:

none

Content:

If the module is not yet initialized and if default development error detection for the Crypto Driver is enabled, the function Crypto_CertificateParse shall report CRYPTO_E_UNINIT to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]

[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]

[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]

[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.80 Specification Item SWS_Crypto_00169

Trace References:

none

Content:

If the parameter cryptoKeyId is out of range and if default development error detection for the Crypto Driver is enabled, the function Crypto_CertificateParse shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

Problem description:

replace "default error detection" with "development error detection"
–Last change on issue 76932 comment 2–

Agreed solution:

SWS_CryIf:
replace "default error" detection with "development error detection" in requirement:
[SWS_CryIf_00016]
[SWS_CryIf_00017]
[SWS_CryIf_00027]
[SWS_CryIf_00028]
[SWS_CryIf_00029]
[SWS_CryIf_00129]
[SWS_CryIf_00130]
[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]

[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]

[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]

[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.81 Specification Item SWS_Crypto_00172

Trace References:

none

Content:

If the module is not yet initialized and if default development error detection for the Crypto Driver is enabled, the function Crypto_CertificateVerify shall report CRYPTO_E_UNINIT to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]

[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]

[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]

[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.82 Specification Item SWS_Crypto_00173

Trace References:

none

Content:

If the parameter cryptoKeyId is out of range and if default development error detection for the Crypto Driver is enabled, the function Crypto_CertificateVerify shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]
  [SWS_CryIf_00056]
  [SWS_CryIf_00057]
  [SWS_CryIf_00059]
  [SWS_CryIf_00060]
  [SWS_CryIf_00062]
  [SWS_CryIf_00063]
  [SWS_CryIf_00064]
  [SWS_CryIf_00110]
  [SWS_CryIf_00111]

[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]

[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]

[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.83 Specification Item SWS_Crypto_00174

Trace References:

none

Content:

If the parameter validateverifyCryptoKeyId is out of range and if default development error detection for the Crypto Driver is enabled, the function Crypto_CertificateVerify shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

Problem description:

replace "default error detection" with "development error detection"
–Last change on issue 76932 comment 2–

Agreed solution:

SWS_CryIf:
replace "default error" detection with "development error detection" in requirement:
[SWS_CryIf_00016]
[SWS_CryIf_00017]
[SWS_CryIf_00027]
[SWS_CryIf_00028]
[SWS_CryIf_00029]
[SWS_CryIf_00129]
[SWS_CryIf_00130]
[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]

[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]

[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]

[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

- RfC #77937: [CRYPTO] Parameters inconsistences in SWS Crypto

  Problem description:

  Hello,
  Please verify the following inconsistencies between naming inside AUTOSAR_SWS_CryptoDriver:

  1.
  [SWS_Crypto_91013] the parameter name is entropyLength
  [SWS_Crypto_00131] the parameter name is seedLength
  Is parameter name seedLength or entropyLength?

  2.
  [SWS_Crypto_91010] the parameter name is partnerPublicValueLength
  [SWS_Crypto_00115] the parameter name is partnerPubValueLength.
  Is parameter name partnerPublicValueLength or partnerPubValueLength?

  3.
  [SWS_Crypto_00171] the parameter name is verifyCryptoKeyId
  [SWS_Crypto_00174] the parameter name is validateCryptoKeyId
  Is parameter name verifyCryptoKeyId or validateCryptoKeyId?

  Thank you,
  Alexandra

  Agreed solution:

  [SWS_Crypto_00130] Replace seedPtr with entropyPtr
  [SWS_Crypto_00131] Replace seedLength with entropyLength.
  [SWS_Crypto_00115] Replace partnerPubValueLength with partnerPublicValue-

Length.
[SWS_Crypto_00174] Replace validateCryptoKeyId with verifyCryptoKeyId
–Last change on issue 77937 comment 3–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.84 Specification Item SWS_Crypto_00175

Trace References:

none

Content:

If the parameter verifyPtr is a null pointer and if default development error detection for the Crypto Driver is enabled, the function Crypto_CertificateVerify shall report CRYPTO_E_PARAM_POINTER to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]
  [SWS_CryIf_00029]
  [SWS_CryIf_00129]
  [SWS_CryIf_00130]
  [SWS_CryIf_00131]
  [SWS_CryIf_00049]
  [SWS_CryIf_00050]
  [SWS_CryIf_00052]
  [SWS_CryIf_00053]

[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]

[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.85 Specification Item SWS_Crypto_00180

Trace References:

none

Content:

If the parameter targetCryptoKeyId is out of range and if development error detection for the Crypto Driver is enabled, the function Crypto_KeyDerive shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76764: Check parameter of Crypto_KeyDerive

  Problem description:

  The parameter "targetCryptoKeyId" of Crypto_KeyDerive inconsistent with each others:
  The range of the first parameter "cryptoKeyId" is considered in [SWS_Crypto_00098].
  But,the second parameter "targetCryptoKeyId" is not considered.
  The range of all parameter should be checked.
  –Last change on issue 76764 comment 5–

  Agreed solution:

  [SWS_Crypto_xxxxx] If the parameter targetCryptoKeyId is out of range and if default
  error detection for the Crypto Driver is enabled, the function Crypto_KeyDerive
  shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.
  ()
  –Last change on issue 76764 comment 2–

  BW-C-Level:

Application	Specification	Bus
1	1	1

- RfC #76932: default error detection -> development error detection

  Problem description:

  replace "default error detection" with "development error detection"
  –Last change on issue 76932 comment 2–

  Agreed solution:

  SWS_CryIf:
  replace "default error" detection with "development error detection" in requirement:
  [SWS_CryIf_00016]
  [SWS_CryIf_00017]
  [SWS_CryIf_00027]
  [SWS_CryIf_00028]

[SWS_CryIf_00029]
[SWS_CryIf_00129]
[SWS_CryIf_00130]
[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]
[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]

[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]
[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]

[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]
[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.86 Specification Item SWS_Crypto_00181

Trace References:

none

Content:

If cancellation of the currently processed is not possible due to limitations, the result of the job shall be discarded and the callback notification shall be suppressed.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77374: Postponed Crypto_CancelJob()

  Problem description:

  If a job cannot be canceled by Crypto_CancelJob() immediately, it is not clear how to proceed. The requirements say:

  [SWS_Crypto_00143] If no errors are detected by Crypto Driver, the service Crypto_CancelJob() shall remove the job from the queue. If the job is currently processed it shall be cancelled. When cancellation of current processing is not possible due to limitations, the result shall be discarded and the callback notification shall be suppressed.

  [SWS_Crypto_00144] If a job is canceled, it shall return CRYPTO_E_JOB_CANCELED either with the callback, when the job is an asynchronous job or as the return value of the function Crypto_CancelJob(), in case the job is synchronous.

  The following questions arise:
  (i) Is it meant in [SWS_Crypto_00143] that (only) the notification of the finished job shall be suppressed?
  (ii) [SWS_Crypto_00144]: There is no return value CRYPTO_E_JOB_CANCELED of Crypto_CancelJob(). So what should be the return value?
  (iii) What does Crypto_CancelJob() return when the cancellation is not possible and it has to be postponed till the job has finished? Crypto_CancelJob() cannot wait till the job has finished.

  Could you pl. clarify these questions?

  Agreed solution:

  Attached to ticket
  –Last change on issue 77374 comment 7–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.87 Specification Item SWS_Crypto_00183

Trace References:

none

Content:

If cancellation of the currently processed is not possible immediately due to limitations, Crypto_CancelJob() shall return with CRYPTO_E_JOB_CANCELED as return value.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77374: Postponed Crypto_CancelJob()

  Problem description:

  If a job cannot be canceled by Crypto_CancelJob() immediately, it is not clear how to proceed. The requirements say:

  [SWS_Crypto_00143] If no errors are detected by Crypto Driver, the service Crypto_CancelJob() shall remove the job from the queue. If the job is currently processed it shall be cancelled. When cancellation of current processing is not possible due to limitations, the result shall be discarded and the callback notification shall be suppressed.

  [SWS_Crypto_00144] If a job is canceled, it shall return CRYPTO_E_JOB_CANCELED either with the callback, when the job is an asynchronous job or as the return value of the function Crypto_CancelJob(), in case the job is synchronous.

  The following questions arise:
  (i) Is it meant in [SWS_Crypto_00143] that (only) the notification of the finished job shall be suppressed?
  (ii) [SWS_Crypto_00144]: There is no return value CRYPTO_E_JOB_CANCELED of Crypto_CancelJob(). So what should be the return value?
  (iii) What does Crypto_CancelJob() return when the cancellation is not possible and it has to be postponed till the job has finished? Crypto_CancelJob() cannot wait till the job has finished.

Could you pl. clarify these questions?

Agreed solution:

Attached to ticket
–Last change on issue 77374 comment 7–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.88 Specification Item SWS_Crypto_00184

Trace References:

SRS_CryptoStack_00008

Content:

Asymmetric key material with identification is specified in accordance to RFC5958 in ASN.1 format. The key material with the format specifier CRYPTO_KE_FORMAT_BIN_IDENT_PRIVATEKEY_ PKCS8 needs to follow this format specification:

OneAsymmetricKey ::= SEQUENCE {

version Version,

KeyAlgorithm KeyAlgorithmIdentifier,

keyMaterial KeyMaterial,

attributes* [0] Attributes OPTIONAL,

...,

[[2: publicKey* [1] PublicKey OPTIONAL]],

...

}

* The optional values for key attributes and the PublicKey are currently not used within the crypto driver and is listed here just for compatibility reason to RFC5958. A driver shall tolerate the provision of this information but doesn't need to evaluate its contents.

The elements have the following meaning:

Version ::= INTEGER { v1(0), v2(1) } (v1, ..., v2)

KeyAlgorithmIdentifier ::= AlgorithmIdentifier

{ PUBLIC-KEY,

{ PrivateKeyAlgorithms } }

KeyMaterial ::= OCTET STRING

– Content varies based on the type of the key and is specified by its AlgorithmIdentifier.

– The KeyAlgorithmIdentifier defines which format specifier for KeyMaterial shall be applied.

AlgorithmIdentifier: A value that identifies the format by its object identifier (OID).

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77661: Definition for asymmetric key formats

  Problem description:

  Name: Armin Happel

  Description/Motivation:
  Currently, the AUTOSAR crypto stack specifies to provide asymmetric key material in PKCS# 8 format only [see SWS_CSM_00951]. However, the standard is not precise enough and defines only the usage of private key material. Optionally, public key material can be provided in addition. This provides the lack of definition in the AUTOSAR stack, that public keys cannot be provided for certain algorithms, such as signature verification.
  This RFC extends the current definition so that also public key material can be provided to the crypto stack.

  Agreed solution:

  See attachment: https://bugzilla.autosar.org/attachment.cgi?id=4617
  –Last change on issue 77661 comment 29–

  BW-C-Level:

Application	Specification	Bus
1	4	1

1.89 Specification Item SWS_Crypto_00185

Trace References:

SRS_CryptoStack_00008

Content:

For CRYPTO_KE_FORMAT_BIN_ RSA_PRIVATEKEY the parameter ’KeyMaterial OCTET STRING’ for RSA private keys is defined according to RFC3447 and has the following contents:

KeyMaterial ::= RSAPrivateKey

RSAPrivateKey ::= SEQUENCE {

version Version,

modulus INTEGER, – n

publicExponent INTEGER, – e

privateExponent INTEGER, – d

prime1 INTEGER, – p

prime2 INTEGER, – q

exponent1 INTEGER, – d mod (p-1)

exponent2 INTEGER, – d mod (q-1)

coefficient INTEGER – (inverse of q) mod p }

Version ::= INTEGER { two-prime(0), multi(1) }

The fields of type RSAPrivateKey have the following meanings:

- version is the version number, for compatibility with future revisions of this document. It shall be 0 for this version of the document.
- modulus is the modulus n.
- publicExponent is the public exponent e.
- privateExponent is the private exponent d.
- prime1 is the prime factor p of n.
- prime2 is the prime factor q of n.
- exponent1 is d mod (p-1).

- exponent2 is d mod (q-1).
- coefficient is the Chinese Remainder Theorem coefficient q-1 mod p.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77661: Definition for asymmetric key formats

  Problem description:

  Name: Armin Happel

  Description/Motivation:
  Currently, the AUTOSAR crypto stack specifies to provide asymmetric key material in PKCS# 8 format only [see SWS_CSM_00951]. However, the standard is not precise enough and defines only the usage of private key material. Optionally, public key material can be provided in addition. This provides the lack of definition in the AUTOSAR stack, that public keys cannot be provided for certain algorithms, such as signature verification.
  This RFC extends the current definition so that also public key material can be provided to the crypto stack.

  Agreed solution:

  See attachment: https://bugzilla.autosar.org/attachment.cgi?id=4617
  –Last change on issue 77661 comment 29–

  BW-C-Level:

Application	Specification	Bus
1	4	1

1.90 Specification Item SWS_Crypto_00186

Trace References:

SRS_CryptoStack_00008

Content:

The RSA public key in the format CRYPTO_KE_FORMAT_BIN _RSA_PUBLICKEY is provided as follows:

RSAPublicKey ::= BIT_STRING {

modulus INTEGER, – n

publicExponent INTEGER, – e

}

The fields of type RSAPublicKey have the following meanings:

- modulus is the modulus n.
- publicExponent is the public exponent e.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77661: Definition for asymmetric key formats

  Problem description:

  Name: Armin Happel

  Description/Motivation:
  Currently, the AUTOSAR crypto stack specifies to provide asymmetric key material in PKCS# 8 format only [see SWS_CSM_00951]. However, the standard is not precise enough and defines only the usage of private key material. Optionally, public key material can be provided in addition. This provides the lack of definition in the AUTOSAR stack, that public keys cannot be provided for certain algorithms, such as signature verification.
  This RFC extends the current definition so that also public key material can be provided to the crypto stack.

  Agreed solution:

  See attachment: https://bugzilla.autosar.org/attachment.cgi?id=4617
  –Last change on issue 77661 comment 29–

  BW-C-Level:

Application	Specification	Bus
1	4	1

1.91 Specification Item SWS_Crypto_00187

Trace References:

SRS_CryptoStack_00008

Content:

The RSA public key in the format CRYPTO_KE_FORMAT_BIN _IDENT_RSA_PUBLICKEY is provided as follows:

PublicKeyInfo ::= SEQUENCE {

KeyAlgorithmIdentifier ::= AlgorithmIdentifier,

publicKey ::= RSAPublicKey

}

Explanation:

Considering RFC5280, section 4.1, the SubjectPublicKeyInfo follows directly the definition described above. Thus, a key type of CRYPTO_KE_FORMAT_BIN_IDENT_PUBLICKEY matches SubjectPublicKeyInfo and CRYPTO_KE_FORMAT_BIN _RSA_PUBLICKEY matches the subjectPublicKey in this definition.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77661: Definition for asymmetric key formats

  Problem description:

  Name: Armin Happel

  Description/Motivation:
  Currently, the AUTOSAR crypto stack specifies to provide asymmetric key material in PKCS# 8 format only [see SWS_CSM_00951]. However, the standard is not precise enough and defines only the usage of private key material. Optionally, public key material can be provided in addition. This provides the lack of definition in the AUTOSAR stack, that public keys cannot be provided for certain algorithms, such as signature verification.
  This RFC extends the current definition so that also public key material can be provided to the crypto stack.

  Agreed solution:

  See attachment: https://bugzilla.autosar.org/attachment.cgi?id=4617
  –Last change on issue 77661 comment 29–

  BW-C-Level:

Application	Specification	Bus
1	4	1

1.92 Specification Item SWS_Crypto_00188

Trace References:

SRS_CryptoStack_00008

Content:

The algorithm identifier for RSA keys shall have the value 1.2.840.113549.1.1.1. This corresponds to the ASN.1 coded OID value "2A 86 48 86 F7 0D 01 01 01". This OID shall be provided whenever an AlgorithmIdentifier for RSA is required. In other words, when a key has the format CRYPTO_KE_FORMAT_BIN_IDENT_PRIVATEKEY_ PKCS8 or CRYPTO_KE_FORMAT_BIN_IDENT_PUBLICKEY and is used for RSA, the Algorithm Identifier must have this value.

Note: In some cases, a NULL value is followed directly to the OID. So, a value that follows directly after this OID in the same sequence is optional and should be tolerated.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77661: Definition for asymmetric key formats

  Problem description:

  Name: Armin Happel

  Description/Motivation:
  Currently, the AUTOSAR crypto stack specifies to provide asymmetric key material in PKCS# 8 format only [see SWS_CSM_00951]. However, the standard is not precise enough and defines only the usage of private key material. Optionally, public key material can be provided in addition. This provides the lack of definition in the AUTOSAR stack, that public keys cannot be provided for certain algorithms, such as signature verification.
  This RFC extends the current definition so that also public key material can be provided to the crypto stack.

  Agreed solution:

  See attachment: https://bugzilla.autosar.org/attachment.cgi?id=4617
  –Last change on issue 77661 comment 29–

  BW-C-Level:

Application	Specification	Bus
1	4	1

1.93 Specification Item SWS_Crypto_00189

Trace References:

SRS_CryptoStack_00008

Content:

Due to a lack of clear and efficient standard definition for ECC keys, key material for ECC is defined as binary information in the format definition of CRYPTO_KE_FORMAT_BIN_OCTET. The length of data depends on the assigned curve operation.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77661: Definition for asymmetric key formats

  Problem description:

  Name: Armin Happel

  Description/Motivation:
  Currently, the AUTOSAR crypto stack specifies to provide asymmetric key material in PKCS# 8 format only [see SWS_CSM_00951]. However, the standard is not precise enough and defines only the usage of private key material. Optionally, public key material can be provided in addition. This provides the lack of definition in the AUTOSAR stack, that public keys cannot be provided for certain algorithms, such as signature verification.
  This RFC extends the current definition so that also public key material can be provided to the crypto stack.

  Agreed solution:

  See attachment: https://bugzilla.autosar.org/attachment.cgi?id=4617
  –Last change on issue 77661 comment 29–

  BW-C-Level:

Application	Specification	Bus
1	4	1

1.94 Specification Item SWS_Crypto_00190

Trace References:

SRS_CryptoStack_00008

Content:

Public keys for NIST and Brainpool ECC curves are provided with their X and Y coordinates:

ECC Public Key = Point X | Point Y.

The points are stored in little endian format.

The number of bytes for the key depends on the implementation of the curve.

Examples:

NIST curve P(256) public key = X(32) | Y(32)

NIST curve P(192) public key = X(24) | Y(24)

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77661: Definition for asymmetric key formats

  Problem description:

  Name: Armin Happel

  Description/Motivation:
  Currently, the AUTOSAR crypto stack specifies to provide asymmetric key material in PKCS# 8 format only [see SWS_CSM_00951]. However, the standard is not precise enough and defines only the usage of private key material. Optionally, public key material can be provided in addition. This provides the lack of definition in the AUTOSAR stack, that public keys cannot be provided for certain algorithms, such as signature verification.
  This RFC extends the current definition so that also public key material can be provided to the crypto stack.

  Agreed solution:

  See attachment: https://bugzilla.autosar.org/attachment.cgi?id=4617
  –Last change on issue 77661 comment 29–

  BW-C-Level:

Application	Specification	Bus
1	4	1

1.95 Specification Item SWS_Crypto_00191

Trace References:

SRS_CryptoStack_00008

Content:

Private keys for NIST and Brainpool ECC curves are provided with their X and Y coordinates and an additional scalar:

ECC Private Key = Point X | Point Y | Scalar.

The points and the scalar are stored in little endian format.

Example:

Brainpool curve P(256) = X(32) | Y(32) | SCALAR(32)

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77661: Definition for asymmetric key formats

  Problem description:

  Name: Armin Happel

  Description/Motivation:
  Currently, the AUTOSAR crypto stack specifies to provide asymmetric key material in PKCS# 8 format only [see SWS_CSM_00951]. However, the standard is not precise enough and defines only the usage of private key material. Optionally, public key material can be provided in addition. This provides the lack of definition in the AUTOSAR stack, that public keys cannot be provided for certain algorithms, such as signature verification.
  This RFC extends the current definition so that also public key material can be provided to the crypto stack.

  Agreed solution:

  See attachment: https://bugzilla.autosar.org/attachment.cgi?id=4617
  –Last change on issue 77661 comment 29–

  BW-C-Level:

Application	Specification	Bus
1	4	1

1.96 Specification Item SWS_Crypto_00192

Trace References:

SRS_CryptoStack_00008

Content:

The public key information for ED25519 contains a point on the curve:

ED25519 Public Key = Point X

The point is stored in little endian format.

Example:

ED25519 Public Key = X(32).

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77661: Definition for asymmetric key formats

  Problem description:

  Name: Armin Happel

  Description/Motivation:
  Currently, the AUTOSAR crypto stack specifies to provide asymmetric key material in PKCS# 8 format only [see SWS_CSM_00951]. However, the standard is not precise enough and defines only the usage of private key material. Optionally, public key material can be provided in addition. This provides the lack of definition in the AUTOSAR stack, that public keys cannot be provided for certain algorithms, such as signature verification.
  This RFC extends the current definition so that also public key material can be provided to the crypto stack.

  Agreed solution:

  See attachment: https://bugzilla.autosar.org/attachment.cgi?id=4617
  –Last change on issue 77661 comment 29–

  BW-C-Level:

Application	Specification	Bus
1	4	1

1.97 Specification Item SWS_Crypto_00193

Trace References:

SRS_CryptoStack_00008

Content:

The private key information for ED25519 contains a random constant and the point X on the curve:

ED25519 Private Key = Seed K | Point X

The point and the seed are stored in little endian format.

Example:

ED25519 Private Key = Seed K(32) | X(32).

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77661: Definition for asymmetric key formats

  Problem description:

  Name: Armin Happel

  Description/Motivation:
  Currently, the AUTOSAR crypto stack specifies to provide asymmetric key material in PKCS# 8 format only [see SWS_CSM_00951]. However, the standard is not precise enough and defines only the usage of private key material. Optionally, public key material can be provided in addition. This provides the lack of definition in the AUTOSAR stack, that public keys cannot be provided for certain algorithms, such as signature verification.
  This RFC extends the current definition so that also public key material can be provided to the crypto stack.

  Agreed solution:

  See attachment: https://bugzilla.autosar.org/attachment.cgi?id=4617
  –Last change on issue 77661 comment 29–

  BW-C-Level:

Application	Specification	Bus
1	4	1

1.98 Specification Item SWS_Crypto_00194

Trace References:

none

Content:

Type of error	Related error code	Value [hex]
Buffer is too small for operation	CRYPTO_E_RE_SMALL_BUFFER	0x00
Requested key is not available	CRYPTO_E_RE_KEY_NOT_AVAILABLE	0x01
Key cannot be read	CRYPTO_E_RE_KEY_READ_FAIL	0x02
Entropy is too low	CRYPTO_E_RE_ENTROPY_EXHAUSTED	0x03

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76636: Rollout of ’Runtime errors’ for entire crypto stack documents

  Problem description:

  Crypto Stack documents are not in line with the RfC # 59085.

  In SWS_secureOnboardCommunication
  Example1: SECOC_E_CRYPTO_FAILURE in the is a development error, but should be a runtime error.

  In SWS_CryptoServiceManager
  Example2: CSM_E_SERVICE_NOT_STARTED is not referenced.

  Example3: CSM_E_PARAM_HANDLE is not referenced in chapter 7.3. It is not clear development error or runtime error.
  –Last change on issue 76636 comment 33–

  Agreed solution:

  CryptoInterface:
  https://bugzilla.autosar.org/attachment.cgi?id=4587

  CryptoServiceManager:
  https://bugzilla.autosar.org/attachment.cgi?id=4614

  CryptoDriver:

https://bugzilla.autosar.org/attachment.cgi?id=4613

SecureOnboardCommunication:
https://bugzilla.autosar.org/attachment.cgi?id=4598
–Last change on issue 76636 comment 41–

BW-C-Level:

Application	Specification	Bus
1	4	1

- RfC #76783: Typo or copy/paste mistakes

  Problem description:

  Hello,

  I found some other mistakes in the specification documents. Most of them are typos or copy/paste mistakes. As document owner of the CryptoServiceManager, I need a confirmation from someone else, before I can implement them into the document.

  AUTOSAR_SWS_CryptoDriver:
  [SWS_Crypto_00139]: CRYPTO_E_KEY_EXTRACT_DENIED does not exist anymore. Replace error code with CRYPTO_E_KEY_READ_FAIL.
  [SWS_Crypto_91005]: Crypto_KeyValidSet() shall be named analogously to Csm_KeySetValid() and CryIf_KeySetValid(). Therefore, rename Crypto_KeyValidSet() to Crypto_KeySetValid().
  [SWS_Crypto_00071]: In table: inputLengthPtr, secondaryInputLengthPtr, tertiaryInputLengthPtr are no pointer anymore. rename them to inputLength, secondaryInputLength, tertiaryInputLength

  AUTOSAR_SWS_CryptoServiceManager:
  [SWS_Csm_01035]: Csm_KeyCopy() shall call CryIf_KeyCopy() not CryIf_KeyElementCopy().
  [SWS_Csm_01080]: Csm_AsymPrivateKeyType is not up-to-date. It should be modified like [SWS_Csm_00076] Csm_AsymPublicKeyType or [SWS_Csm_01082] Csm_SymKeyType.
  SWS_Csm_00455
  [SWS_Csm_00455]: tag as obsolete
  [ECUC_Csm_00188]: typo: CsmMacGenerateAlgorithmFamiliy -> CsmMacGenerateAlgorithmFamily
  [ECUC_Csm_00049]: CsmMacVerifyAlgorithmMode missing. (see analogues CsmMacGenerateAlgorithmMode [ECUC_Csm_00189])

[SWS_Csm_00966]: CopyPaste mistake: Delete: "Wrong return values - here are the correct ones:"
[SWS_Csm_01023]: plaintextLength description wrong. replace with "Contains the number of bytes to encrypt."
[SWS_Csm_01023]: typo "associtatedDataLengthPtr" and it is no pointer. replace with: "associatedDataLength"
[SWS_Csm_01025]: typo, replace line "job->jobPrimitiveInputOutput.outputLength = ciphertextLength," with
"job->jobPrimitiveInputOutput.outputLengthPtr = ciphertextLengthPtr,"
[SWS_Csm_01013]: typo: rename "PrimitiveInputOutput" to "jobPrimitiveInputOutput". Or is this rename intended?
Then every assignment of "jobPrimitiveInputOutput" has to be renamed to "primitiveInputOutput" like
"job->jobPrimitiveInputOutput.mode = mode," has to be modified to "job->primitiveInputOutput.mode = mode,"
[SWS_Csm_01026]: typo: replace "associtatedDataLength" with "associatedDataLength"
[SWS_Csm_01027]: missing line: "job->jobPrimitiveInputOutput.verifyPtr = verifyPtr."
[SWS_Csm_00992]: copypaste mistake: replace "mode: The Crypto_JobInfoType job with the corresponding jobId shall be modified in the following way:" with ""mode: Indicates which operation mode(s) to perfom."
[SWS_Csm_00992]: copypaste mistake: replace "resultLengthPtr: Contains the number of bytes of the associated data." with ""resultLengthPtr: Holds a pointer to the memory location in which the output length in bytes of the signature is stored. On calling this function, this parameter shall contain the size of the buffer provided by resultPtr. When the request has finished, the actual length of the returned value shall be stored."
[SWS_Csm_01543]: description wrong. replace with "Generate a random number and stores it in the memory location pointed by the result pointer."
[SWS_Csm_00168]: description wrong, there is no IV. replace with "This function is deprecated. Sets the key for symmetrical encryption."
[SWS_Csm_01031]: description wrong, it is not decrement. "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterRead Service"

A proposed solution is added, too.

Agreed solution:

AUTOSAR_SWS_CryptoDriver:
[SWS_Crypto_00139]: Replace CRYPTO_E_KEY_EXTRACT_DENIED with CRYPTO_E_KEY_READ_FAIL.
[SWS_CryIf_91015]: Remove CRYPTO_E_KEY_EXTRACT_DENIED
[SWS_Crypto_91005]: Set Crypto_KeyValidSet obsolete.

[SWS_Crypto_xxx]: Add Crypto_KeySetValid as API (Description according to SWS_Crypto_91005)
[SWS_Crypto_00082]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add E_UNINIT DET check SWS for Crypto_KeySetValid (Text according to SWS_Crypto_00082)
[SWS_Crypto_00083]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add E_PARAM_HANDLE DET check SWS for Crypto_KeySetValid (Text according to SWS_Crypto_00083)
last sentence in 8.2.4.1.2: Rename Crypto_KeyValidSet to Crypto_KeySetValid

[SWS_Crypto_00071]: rename inputLengthPtr, secondaryInputLengthPtr, tertiaryInputLengthPtr to inputLength, secondaryInputLength, tertiaryInputLength

AUTOSAR_SWS_CryptoServiceManager:
[SWS_Csm_01035]: CryIf_KeyElementCopy() shall be replaced with CryIf_KeyCopy().

[SWS_Csm_01080]: replace with (see [SWS_Csm_00076]):
Name: Csm_AsymPrivateKeyType
Kind: Structure
Elements:
length: uint32: This element contains the length in bytes of the key stored in element ’data’
data: Csm_AsymPrivateKeyArrayType: This element contains the key data or a key handle.
Description: Structure for the private asymmetrical key.
Variation: –

[SWS_Csm_00455]: tag as obsolete

[ECUC_Csm_00188]: typo: CsmMacGenerateAlgorithmFamiliy -> CsmMacGenerateAlgorithmFamily
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmMode (see analogues CsmMacGenerateAlgorithmMode [ECUC_Csm_00189])
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmModeCustom (see analogues CsmMacGenerateAlgorithmModeCustom [ECUC_Csm_00189])
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmKeyLength (see analogues CsmMacGenerateAlgorithmKeyLength [ECUC_Csm_00189])
[SWS_Csm_00966]: Delete: "Wrong return values - here are the correct ones:"
[SWS_Csm_01023]: Replace description with: "Contains the number of bytes to encrypt."
[SWS_Csm_01023]: Replace "associatedDataLengthPtr" with "associatedDataLength"

[SWS_Csm_01025]: Replace line "job->jobPrimitiveInputOutput.outputLength = ciphertextLength," with
"job->jobPrimitiveInputOutput.outputLengthPtr = ciphertextLengthPtr,"
[SWS_Csm_01013]: rename "PrimitiveInputOutput" to "jobPrimitiveInputOutput". rename "state" to "jobState".
[SWS_Csm_01026]: replace "associtatedDataLength" with "associatedDataLength"
[SWS_Csm_01027]: add line: "job->jobPrimitiveInputOutput.verifyPtr = verifyPtr."
[SWS_Csm_00992]: replace "mode: The Crypto_JobInfoType job with the corresponding jobId shall be modified in the following way:" with ""mode: Indicates which operation mode(s) to perform."
[SWS_Csm_00992]: replace "resultLengthPtr: Contains the number of bytes of the associated data." with ""resultLengthPtr: Holds a pointer to the memory location in which the output length in bytes of the signature is stored. On calling this function, this parameter shall contain the size of the buffer provided by resultPtr. When the request has finished, the actual length of the returned value shall be stored."
[SWS_Csm_01543]: replace description with "Generate a random number and stores it in the memory location pointed by the result pointer."
[SWS_Csm_00168]: replace description with "This function is deprecated. Sets the key for symmetrical encryption."
[SWS_Csm_01031]: replace "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterDecrement Service" with "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterRead Service"
–Last change on issue 76783 comment 29–

BW-C-Level:

Application	Specification	Bus
4	3	1

1.99 Specification Item SWS_Crypto_00195

Trace References:

none

Content:

If a Crypto API is called with a buffer too small to perform the desires operation CRYPTO_E_RE_SMALL_BUFFER shall be reported to the DET and the operation shall not be performed.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76932: default error detection -> development error detection

Problem description:

replace "default error detection" with "development error detection"
–Last change on issue 76932 comment 2–

Agreed solution:

SWS_CryIf:
replace "default error" detection with "development error detection" in requirement:
[SWS_CryIf_00016]
[SWS_CryIf_00017]
[SWS_CryIf_00027]
[SWS_CryIf_00028]
[SWS_CryIf_00029]
[SWS_CryIf_00129]
[SWS_CryIf_00130]
[SWS_CryIf_00131]
[SWS_CryIf_00049]
[SWS_CryIf_00050]
[SWS_CryIf_00052]
[SWS_CryIf_00053]
[SWS_CryIf_00056]
[SWS_CryIf_00057]
[SWS_CryIf_00059]
[SWS_CryIf_00060]
[SWS_CryIf_00062]
[SWS_CryIf_00063]
[SWS_CryIf_00064]
[SWS_CryIf_00110]
[SWS_CryIf_00111]
[SWS_CryIf_00112]
[SWS_CryIf_00116]
[SWS_CryIf_00117]
[SWS_CryIf_00118]
[SWS_CryIf_00068]
[SWS_CryIf_00069]
[SWS_CryIf_00070]
[SWS_CryIf_00071]
[SWS_CryIf_00073]
[SWS_CryIf_00074]
[SWS_CryIf_00076]
[SWS_CryIf_00077]
[SWS_CryIf_00122]
[SWS_CryIf_00122]

[SWS_CryIf_00082]
[SWS_CryIf_00083]
[SWS_CryIf_00084]
[SWS_CryIf_00085]
[SWS_CryIf_00086]
[SWS_CryIf_00090]
[SWS_CryIf_00091]
[SWS_CryIf_00092]
[SWS_CryIf_00093]
[SWS_CryIf_00094]
[SWS_CryIf_00098]
[SWS_CryIf_00099]
[SWS_CryIf_00123]
[SWS_CryIf_00124]
[SWS_CryIf_00125]
[SWS_CryIf_00126]
[SWS_CryIf_00127]
[SWS_CryIf_00107]
[SWS_CryIf_00108]

SWS_Crypto:
replace "default error" detection with "development error detection" in requirement:
[SWS_Crypto_00047]
[SWS_Crypto_00057]
[SWS_Crypto_00058]
[SWS_Crypto_00059]
[SWS_Crypto_00064]
[SWS_Crypto_00067]
[SWS_Crypto_00070]
[SWS_Crypto_00142]
[SWS_Crypto_00136]
[SWS_Crypto_00137]
[SWS_Crypto_00141]
[SWS_Crypto_00123]
[SWS_Crypto_00124]
[SWS_Crypto_00125]
[SWS_Crypto_00075]
[SWS_Crypto_00076]
[SWS_Crypto_00077]
[SWS_Crypto_00078]
[SWS_Crypto_00079]
[SWS_Crypto_00082]
[SWS_Crypto_00083]

[SWS_Crypto_00140]
[SWS_Crypto_00138]
[SWS_Crypto_00085]
[SWS_Crypto_00086]
[SWS_Crypto_00087]
[SWS_Crypto_00088]
[SWS_Crypto_00089]
[SWS_Crypto_00090]
[SWS_Crypto_00093]
[SWS_Crypto_00149]
[SWS_Crypto_00150]
[SWS_Crypto_00151]
[SWS_Crypto_00152]
[SWS_Crypto_00153]
[SWS_Crypto_00156]
[SWS_Crypto_00157]
[SWS_Crypto_00158]
[SWS_Crypto_00161]
[SWS_Crypto_00162]
[SWS_Crypto_00163]
[SWS_Crypto_00164]
[SWS_Crypto_00128]
[SWS_Crypto_00129]
[SWS_Crypto_00130]
[SWS_Crypto_00131]
[SWS_Crypto_00094]
[SWS_Crypto_00095]
[SWS_Crypto_00097]
[SWS_Crypto_00098]
[SWS_Crypto_00103]
[SWS_Crypto_00104]
[SWS_Crypto_00105]
[SWS_Crypto_00106]
[SWS_Crypto_00107]
[SWS_Crypto_00110]
[SWS_Crypto_00111]
[SWS_Crypto_00112]
[SWS_Crypto_00113]
[SWS_Crypto_00115]
[SWS_Crypto_00168]
[SWS_Crypto_00169]
[SWS_Crypto_00172]
[SWS_Crypto_00173]

[SWS_Crypto_00174]
[SWS_Crypto_00175]

SRS_Crypto:
[SRS_CryptoStack_00087] The CSM module shall report detected
development errors to the Development Error Tracer
–>Default Error Tracer
–Last change on issue 76932 comment 2–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.100 Specification Item SWS_Crypto_00196

Trace References:

none

Content:

If the module is not yet initialized and development error detection for the Crypto Driver is enabled, the function Crypto_KeySetValid shall report CRYPTO_E_UNINIT to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76783: Typo or copy/paste mistakes

  Problem description:

  Hello,

  I found some other mistakes in the specification documents. Most of them are typos or copy/paste mistakes. As document owner of the CryptoServiceManager, I need a confirmation from someone else, before I can implement them into the document.

  AUTOSAR_SWS_CryptoDriver:
  [SWS_Crypto_00139]: CRYPTO_E_KEY_EXTRACT_DENIED does not exist anymore. Replace error code with CRYPTO_E_KEY_READ_FAIL.
  [SWS_Crypto_91005]: Crypto_KeyValidSet() shall be named analogously to Csm_KeySetValid() and CryIf_KeySetValid(). Therefore, rename Crypto_KeyValidSet() to Crypto_KeySetValid().

[SWS_Crypto_00071]: In table: inputLengthPtr, secondaryInputLengthPtr, tertiaryInputLengthPtr are no pointer anymore. rename them to inputLength, secondaryInputLength, tertiaryInputLength

AUTOSAR_SWS_CryptoServiceManager:
[SWS_Csm_01035]: Csm_KeyCopy() shall call CryIf_KeyCopy() not CryIf_KeyElementCopy().
[SWS_Csm_01080]: Csm_AsymPrivateKeyType is not up-to-date. It should be modified like [SWS_Csm_00076] Csm_AsymPublicKeyType or [SWS_Csm_01082] Csm_SymKeyType.
SWS_Csm_00455
[SWS_Csm_00455]: tag as obsolete
[ECUC_Csm_00188]: typo: CsmMacGenerateAlgorithmFamiliy -> CsmMacGenerateAlgorithmFamily
[ECUC_Csm_00049]: CsmMacVerifyAlgorithmMode missing. (see analogues CsmMacGenerateAlgorithmMode [ECUC_Csm_00189])
[SWS_Csm_00966]: CopyPaste mistake: Delete: "Wrong return values - here are the correct ones:"
[SWS_Csm_01023]: plaintextLength description wrong. replace with "Contains the number of bytes to encrypt."
[SWS_Csm_01023]: typo "associtatedDataLengthPtr" and it is no pointer. replace with: "associatedDataLength"
[SWS_Csm_01025]: typo, replace line "job->jobPrimitiveInputOutput.outputLength = ciphertextLength," with
"job->jobPrimitiveInputOutput.outputLengthPtr = ciphertextLengthPtr,"
[SWS_Csm_01013]: typo: rename "PrimitiveInputOutput" to "jobPrimitiveInputOutput". Or is this rename intended?
Then every assignment of "jobPrimitiveInputOutput" has to be renamed to "primitiveInputOutput" like
"job->jobPrimitiveInputOutput.mode = mode," has to be modified to "job->primitiveInputOutput.mode = mode,"
[SWS_Csm_01026]: typo: replace "associtatedDataLength" with "associatedDataLength"
[SWS_Csm_01027]: missing line: "job->jobPrimitiveInputOutput.verifyPtr = verifyPtr."
[SWS_Csm_00992]: copypaste mistake: replace "mode: The Crypto_JobInfoType job with the corresponding jobId shall be modified in the following way:" with ""mode: Indicates which operation mode(s) to perfom."
[SWS_Csm_00992]: copypaste mistake: replace "resultLengthPtr: Contains the number of bytes of the associated data." with ""resultLengthPtr: Holds a pointer to the memory location in which the output length in bytes of the signature is stored. On calling this function, this parameter shall contain the size of the buffer provided

by resultPtr. When the request has finished, the actual length of the returned value shall be stored."
[SWS_Csm_01543]: description wrong. replace with "Generate a random number and stores it in the memory location pointed by the result pointer."
[SWS_Csm_00168]: description wrong, there is no IV. replace with "This function is deprecated. Sets the key for symmetrical encryption."
[SWS_Csm_01031]: description wrong, it is not decrement. "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterRead Service"

A proposed solution is added, too.

Agreed solution:

AUTOSAR_SWS_CryptoDriver:
[SWS_Crypto_00139]: Replace CRYPTO_E_KEY_EXTRACT_DENIED with CRYPTO_E_KEY_READ_FAIL.
[SWS_CryIf_91015]: Remove CRYPTO_E_KEY_EXTRACT_DENIED
[SWS_Crypto_91005]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add Crypto_KeySetValid as API (Description according to SWS_Crypto_91005)
[SWS_Crypto_00082]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add E_UNINIT DET check SWS for Crypto_KeySetValid (Text according to SWS_Crypto_00082)
[SWS_Crypto_00083]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add E_PARAM_HANDLE DET check SWS for Crypto_KeySetValid (Text according to SWS_Crypto_00083)
last sentence in 8.2.4.1.2: Rename Crypto_KeyValidSet to Crypto_KeySetValid

[SWS_Crypto_00071]: rename inputLengthPtr, secondaryInputLengthPtr, tertiaryInputLengthPtr to inputLength, secondaryInputLength, tertiaryInputLength

AUTOSAR_SWS_CryptoServiceManager:
[SWS_Csm_01035]: CryIf_KeyElementCopy() shall be replaced with CryIf_KeyCopy().

[SWS_Csm_01080]: replace with (see [SWS_Csm_00076]):
Name: Csm_AsymPrivateKeyType
Kind: Structure
Elements:
length: uint32: This element contains the length in bytes of the key stored in element 'data'
data: Csm_AsymPrivateKeyArrayType: This element contains the key data or a key handle.
Description: Structure for the private asymmetrical key.

Variation: –

[SWS_Csm_00455]: tag as obsolete

[ECUC_Csm_00188]: typo: CsmMacGenerateAlgorithmFamiliy -> CsmMacGenerateAlgorithmFamily
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmMode (see analogues CsmMacGenerateAlgorithmMode [ECUC_Csm_00189])
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmModeCustom (see analogues CsmMacGenerateAlgorithmModeCustom [ECUC_Csm_00189])
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmKeyLength (see analogues CsmMacGenerateAlgorithmKeyLength [ECUC_Csm_00189])
[SWS_Csm_00966]: Delete: "Wrong return values - here are the correct ones:"
[SWS_Csm_01023]: Replace description with: "Contains the number of bytes to encrypt."
[SWS_Csm_01023]: Replace "associatedDataLengthPtr" with "associatedDataLength"
[SWS_Csm_01025]: Replace line "job->jobPrimitiveInputOutput.outputLength = ciphertextLength," with
"job->jobPrimitiveInputOutput.outputLengthPtr = ciphertextLengthPtr,"
[SWS_Csm_01013]: rename "PrimitiveInputOutput" to "jobPrimitiveInputOutput". rename "state" to "jobState".
[SWS_Csm_01026]: replace "associtatedDataLength" with "associatedDataLength"
[SWS_Csm_01027]: add line: "job->jobPrimitiveInputOutput.verifyPtr = verifyPtr."
[SWS_Csm_00992]: replace "mode: The Crypto_JobInfoType job with the corresponding jobId shall be modified in the following way:" with ""mode: Indicates which operation mode(s) to perform."
[SWS_Csm_00992]: replace "resultLengthPtr: Contains the number of bytes of the associated data." with ""resultLengthPtr: Holds a pointer to the memory location in which the output length in bytes of the signature is stored. On calling this function, this parameter shall contain the size of the buffer provided by resultPtr. When the request has finished, the actual length of the returned value shall be stored."
[SWS_Csm_01543]: replace description with "Generate a random number and stores it in the memory location pointed by the result pointer."
[SWS_Csm_00168]: replace description with "This function is deprecated. Sets the key for symmetrical encryption."
[SWS_Csm_01031]: replace "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterDecrement Service" with "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterRead Service"
–Last change on issue 76783 comment 29–

BW-C-Level:

Application	Specification	Bus
4	3	1

1.101 Specification Item SWS_Crypto_00197

Trace References:

none

Content:

If parameter cryptoKeyId is out of range and if development error detection for the Crypto Driver is enabled, the function Crypto_KeySetValid shall report CRYPTO_E_PARAM_HANDLE to the DET and return E_NOT_OK.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76783: Typo or copy/paste mistakes

  Problem description:

  Hello,

  I found some other mistakes in the specification documents. Most of them are typos or copy/paste mistakes. As document owner of the CryptoServiceManager, I need a confirmation from someone else, before I can implement them into the document.

  AUTOSAR_SWS_CryptoDriver:
  [SWS_Crypto_00139]: CRYPTO_E_KEY_EXTRACT_DENIED does not exist anymore. Replace error code with CRYPTO_E_KEY_READ_FAIL.
  [SWS_Crypto_91005]: Crypto_KeyValidSet() shall be named analogously to Csm_KeySetValid() and CryIf_KeySetValid(). Therefore, rename Crypto_KeyValidSet() to Crypto_KeySetValid().
  [SWS_Crypto_00071]: In table: inputLengthPtr, secondaryInputLengthPtr, tertiaryInputLengthPtr are no pointer anymore. rename them to inputLength, secondaryInputLength, tertiaryInputLength

  AUTOSAR_SWS_CryptoServiceManager:
  [SWS_Csm_01035]: Csm_KeyCopy() shall call CryIf_KeyCopy() not CryIf_KeyElementCopy().
  [SWS_Csm_01080]: Csm_AsymPrivateKeyType is not up-to-date. It should be modified like [SWS_Csm_00076] Csm_AsymPublicKeyType or [SWS_Csm_01082]

Csm_SymKeyType.
SWS_Csm_00455
[SWS_Csm_00455]: tag as obsolete
[ECUC_Csm_00188]: typo: CsmMacGenerateAlgorithmFamiliy -> CsmMacGenerateAlgorithmFamily
[ECUC_Csm_00049]: CsmMacVerifyAlgorithmMode missing. (see analogues CsmMacGenerateAlgorithmMode [ECUC_Csm_00189])
[SWS_Csm_00966]: CopyPaste mistake: Delete: "Wrong return values - here are the correct ones:"
[SWS_Csm_01023]: plaintextLength description wrong. replace with "Contains the number of bytes to encrypt."
[SWS_Csm_01023]: typo "associtatedDataLengthPtr" and it is no pointer. replace with: "associatedDataLength"
[SWS_Csm_01025]: typo, replace line "job->jobPrimitiveInputOutput.outputLength = ciphertextLength," with
"job->jobPrimitiveInputOutput.outputLengthPtr = ciphertextLengthPtr,"
[SWS_Csm_01013]: typo: rename "PrimitiveInputOutput" to "jobPrimitiveInputOutput". Or is this rename intended?
Then every assignment of "jobPrimitiveInputOutput" has to be renamed to "primitiveInputOutput" like
"job->jobPrimitiveInputOutput.mode = mode," has to be modified to "job->primitiveInputOutput.mode = mode,"
[SWS_Csm_01026]: typo: replace "associtatedDataLength" with "associatedDataLength"
[SWS_Csm_01027]: missing line: "job->jobPrimitiveInputOutput.verifyPtr = verifyPtr."
[SWS_Csm_00992]: copypaste mistake: replace "mode: The Crypto_JobInfoType job with the corresponding jobId shall be modified in the following way:" with ""mode: Indicates which operation mode(s) to perfom."
[SWS_Csm_00992]: copypaste mistake: replace "resultLengthPtr: Contains the number of bytes of the associated data." with ""resultLengthPtr: Holds a pointer to the memory location in which the output length in bytes of the signature is stored. On calling this function, this parameter shall contain the size of the buffer provided by resultPtr. When the request has finished, the actual length of the returned value shall be stored."
[SWS_Csm_01543]: description wrong. replace with "Generate a random number and stores it in the memory location pointed by the result pointer."
[SWS_Csm_00168]: description wrong, there is no IV. replace with "This function is deprecated. Sets the key for symmetrical encryption."
[SWS_Csm_01031]: description wrong, it is not decrement. "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterRead Service"

A proposed solution is added, too.

Agreed solution:

AUTOSAR_SWS_CryptoDriver:
[SWS_Crypto_00139]: Replace CRYPTO_E_KEY_EXTRACT_DENIED with CRYPTO_E_KEY_READ_FAIL.
[SWS_CryIf_91015]: Remove CRYPTO_E_KEY_EXTRACT_DENIED
[SWS_Crypto_91005]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add Crypto_KeySetValid as API (Description according to SWS_Crypto_91005)
[SWS_Crypto_00082]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add E_UNINIT DET check SWS for Crypto_KeySetValid (Text according to SWS_Crypto_00082)
[SWS_Crypto_00083]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add E_PARAM_HANDLE DET check SWS for Crypto_KeySetValid (Text according to SWS_Crypto_00083)
last sentence in 8.2.4.1.2: Rename Crypto_KeyValidSet to Crypto_KeySetValid

[SWS_Crypto_00071]: rename inputLengthPtr, secondaryInputLengthPtr, tertiaryInputLengthPtr to inputLength, secondaryInputLength, tertiaryInputLength

AUTOSAR_SWS_CryptoServiceManager:
[SWS_Csm_01035]: CryIf_KeyElementCopy() shall be replaced with CryIf_KeyCopy().

[SWS_Csm_01080]: replace with (see [SWS_Csm_00076]):
Name: Csm_AsymPrivateKeyType
Kind: Structure
Elements:
length: uint32: This element contains the length in bytes of the key stored in element 'data'
data: Csm_AsymPrivateKeyArrayType: This element contains the key data or a key handle.
Description: Structure for the private asymmetrical key.
Variation: –

[SWS_Csm_00455]: tag as obsolete

[ECUC_Csm_00188]: typo: CsmMacGenerateAlgorithmFamiliy -> CsmMacGenerateAlgorithmFamily
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmMode (see analogues CsmMacGenerateAlgorithmMode [ECUC_Csm_00189])
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmModeCustom (see analogues CsmMacGenerateAlgorithmModeCustom [ECUC_Csm_00189])

[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmKeyLength (see analogues CsmMacGenerateAlgorithmKeyLength [ECUC_Csm_00189])
[SWS_Csm_00966]: Delete: "Wrong return values - here are the correct ones:"
[SWS_Csm_01023]: Replace description with: "Contains the number of bytes to encrypt."
[SWS_Csm_01023]: Replace "associatedDataLengthPtr" with "associatedDataLength"
[SWS_Csm_01025]: Replace line "job->jobPrimitiveInputOutput.outputLength = ciphertextLength," with
"job->jobPrimitiveInputOutput.outputLengthPtr = ciphertextLengthPtr,"
[SWS_Csm_01013]: rename "PrimitiveInputOutput" to "jobPrimitiveInputOutput". rename "state" to "jobState".
[SWS_Csm_01026]: replace "associtatedDataLength" with "associatedDataLength"
[SWS_Csm_01027]: add line: "job->jobPrimitiveInputOutput.verifyPtr = verifyPtr."
[SWS_Csm_00992]: replace "mode: The Crypto_JobInfoType job with the corresponding jobId shall be modified in the following way:" with ""mode: Indicates which operation mode(s) to perform."
[SWS_Csm_00992]: replace "resultLengthPtr: Contains the number of bytes of the associated data." with ""resultLengthPtr: Holds a pointer to the memory location in which the output length in bytes of the signature is stored. On calling this function, this parameter shall contain the size of the buffer provided by resultPtr. When the request has finished, the actual length of the returned value shall be stored."
[SWS_Csm_01543]: replace description with "Generate a random number and stores it in the memory location pointed by the result pointer."
[SWS_Csm_00168]: replace description with "This function is deprecated. Sets the key for symmetrical encryption."
[SWS_Csm_01031]: replace "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterDecrement Service" with "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterRead Service"
–Last change on issue 76783 comment 29–

BW-C-Level:

Application	Specification	Bus
4	3	1

1.102 Specification Item SWS_Crypto_91005

Trace References:

none

Content:

Service name:	Crypto_KeyValidSet (obsolete)Crypto_KeyValidSet	
Syntax:	Std_ReturnType Crypto_KeyValidSet(uint32 cryptoKeyId)	
Service ID[hex]:	0x05	
Sync/Async:	Synchronous	
Reentrancy:	Non Reentrant	
Parameters (in):	cryptoKeyIdCrypto_KeyValidSet.cryptoKeyId	Holds the identifier of the key which 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:	Sets the key state of the key identified by cryptoKeyId to valid. Tags: atp.Status=obsolete	

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76783: Typo or copy/paste mistakes

  Problem description:

  Hello,

  I found some other mistakes in the specification documents. Most of them are typos or copy/paste mistakes. As document owner of the CryptoServiceManager, I need a confirmation from someone else, before I can implement them into the document.

  AUTOSAR_SWS_CryptoDriver:
  [SWS_Crypto_00139]: CRYPTO_E_KEY_EXTRACT_DENIED does not exist anymore. Replace error code with CRYPTO_E_KEY_READ_FAIL.
  [SWS_Crypto_91005]: Crypto_KeyValidSet() shall be named analogously to Csm_KeySetValid() and CryIf_KeySetValid(). Therefore, rename Crypto_KeyValidSet() to Crypto_KeySetValid().
  [SWS_Crypto_00071]: In table: inputLengthPtr, secondaryInputLengthPtr, tertiaryInputLengthPtr are no pointer anymore. rename them to inputLength, secondaryInputLength, tertiaryInputLength

  AUTOSAR_SWS_CryptoServiceManager:
  [SWS_Csm_01035]: Csm_KeyCopy() shall call CryIf_KeyCopy() not

CryIf_KeyElementCopy().
[SWS_Csm_01080]: Csm_AsymPrivateKeyType is not up-to-date. It should be modified like [SWS_Csm_00076] Csm_AsymPublicKeyType or [SWS_Csm_01082] Csm_SymKeyType.
SWS_Csm_00455
[SWS_Csm_00455]: tag as obsolete
[ECUC_Csm_00188]: typo: CsmMacGenerateAlgorithmFamiliy -> CsmMacGenerateAlgorithmFamily
[ECUC_Csm_00049]: CsmMacVerifyAlgorithmMode missing. (see analogues CsmMacGenerateAlgorithmMode [ECUC_Csm_00189])
[SWS_Csm_00966]: CopyPaste mistake: Delete: "Wrong return values - here are the correct ones:"
[SWS_Csm_01023]: plaintextLength description wrong. replace with "Contains the number of bytes to encrypt."
[SWS_Csm_01023]: typo "associtatedDataLengthPtr" and it is no pointer. replace with: "associatedDataLength"
[SWS_Csm_01025]: typo, replace line "job->jobPrimitiveInputOutput.outputLength = ciphertextLength," with
"job->jobPrimitiveInputOutput.outputLengthPtr = ciphertextLengthPtr,"
[SWS_Csm_01013]: typo: rename "PrimitiveInputOutput" to "jobPrimitiveInputOutput". Or is this rename intended?
Then every assignment of "jobPrimitiveInputOutput" has to be renamed to "primitiveInputOutput" like
"job->jobPrimitiveInputOutput.mode = mode," has to be modified to "job->primitiveInputOutput.mode = mode,"
[SWS_Csm_01026]: typo: replace "associtatedDataLength" with "associatedDataLength"
[SWS_Csm_01027]: missing line: "job->jobPrimitiveInputOutput.verifyPtr = verifyPtr."
[SWS_Csm_00992]: copypaste mistake: replace "mode: The Crypto_JobInfoType job with the corresponding jobId shall be modified in the following way:" with ""mode: Indicates which operation mode(s) to perfom."
[SWS_Csm_00992]: copypaste mistake: replace "resultLengthPtr: Contains the number of bytes of the associated data." with ""resultLengthPtr: Holds a pointer to the memory location in which the output length in bytes of the signature is stored. On calling this function, this parameter shall contain the size of the buffer provided by resultPtr. When the request has finished, the actual length of the returned value shall be stored."
[SWS_Csm_01543]: description wrong. replace with "Generate a random number and stores it in the memory location pointed by the result pointer."
[SWS_Csm_00168]: description wrong, there is no IV. replace with "This function is deprecated. Sets the key for symmetrical encryption."
[SWS_Csm_01031]: description wrong, it is not decrement.

"CRYPTO_SECCOUNTERREAD 0x0A SecureCounterRead Service"

A proposed solution is added, too.

Agreed solution:

AUTOSAR_SWS_CryptoDriver:
[SWS_Crypto_00139]: Replace CRYPTO_E_KEY_EXTRACT_DENIED with CRYPTO_E_KEY_READ_FAIL.
[SWS_CryIf_91015]: Remove CRYPTO_E_KEY_EXTRACT_DENIED
[SWS_Crypto_91005]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add Crypto_KeySetValid as API (Description according to SWS_Crypto_91005)
[SWS_Crypto_00082]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add E_UNINIT DET check SWS for Crypto_KeySetValid (Text according to SWS_Crypto_00082)
[SWS_Crypto_00083]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add E_PARAM_HANDLE DET check SWS for Crypto_KeySetValid (Text according to SWS_Crypto_00083)
last sentence in 8.2.4.1.2: Rename Crypto_KeyValidSet to Crypto_KeySetValid

[SWS_Crypto_00071]: rename inputLengthPtr, secondaryInputLengthPtr, tertiaryInputLengthPtr to inputLength, secondaryInputLength, tertiaryInputLength

AUTOSAR_SWS_CryptoServiceManager:
[SWS_Csm_01035]: CryIf_KeyElementCopy() shall be replaced with CryIf_KeyCopy().

[SWS_Csm_01080]: replace with (see [SWS_Csm_00076]):
Name: Csm_AsymPrivateKeyType
Kind: Structure
Elements:
length: uint32: This element contains the length in bytes of the key stored in element 'data'
data: Csm_AsymPrivateKeyArrayType: This element contains the key data or a key handle.
Description: Structure for the private asymmetrical key.
Variation: –

[SWS_Csm_00455]: tag as obsolete

[ECUC_Csm_00188]: typo: CsmMacGenerateAlgorithmFamiliy -> CsmMacGenerateAlgorithmFamily
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmMode (see analogues CsmMac-

GenerateAlgorithmMode [ECUC_Csm_00189])
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmModeCustom (see analogues CsmMacGenerateAlgorithmModeCustom [ECUC_Csm_00189])
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmKeyLength (see analogues CsmMacGenerateAlgorithmKeyLength [ECUC_Csm_00189])
[SWS_Csm_00966]: Delete: "Wrong return values - here are the correct ones:"
[SWS_Csm_01023]: Replace description with: "Contains the number of bytes to encrypt."
[SWS_Csm_01023]: Replace "associatedDataLengthPtr" with "associatedDataLength"
[SWS_Csm_01025]: Replace line "job->jobPrimitiveInputOutput.outputLength = ciphertextLength," with
"job->jobPrimitiveInputOutput.outputLengthPtr = ciphertextLengthPtr,"
[SWS_Csm_01013]: rename "PrimitiveInputOutput" to "jobPrimitiveInputOutput". rename "state" to "jobState".
[SWS_Csm_01026]: replace "associtatedDataLength" with "associatedDataLength"
[SWS_Csm_01027]: add line: "job->jobPrimitiveInputOutput.verifyPtr = verifyPtr."
[SWS_Csm_00992]: replace "mode: The Crypto_JobInfoType job with the corresponding jobId shall be modified in the following way:" with ""mode: Indicates which operation mode(s) to perform."
[SWS_Csm_00992]: replace "resultLengthPtr: Contains the number of bytes of the associated data." with ""resultLengthPtr: Holds a pointer to the memory location in which the output length in bytes of the signature is stored. On calling this function, this parameter shall contain the size of the buffer provided by resultPtr. When the request has finished, the actual length of the returned value shall be stored."
[SWS_Csm_01543]: replace description with "Generate a random number and stores it in the memory location pointed by the result pointer."
[SWS_Csm_00168]: replace description with "This function is deprecated. Sets the key for symmetrical encryption."
[SWS_Csm_01031]: replace "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterDecrement Service" with "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterRead Service"
–Last change on issue 76783 comment 29–

BW-C-Level:

Application	Specification	Bus
4	3	1

1.103 Specification Item SWS_Crypto_91013

Trace References:

none

Content:

Service name:	Crypto_RandomSeedCrypto_RandomSeed	
Syntax:	Std_ReturnType Crypto_RandomSeed(uint32 cryptoKeyId, const uint8* entropyseedPtr, uint32 entropyseedLength)	
Service ID[hex]:	0x0d	
Sync/Async:	Synchronous	
Reentrancy:	Reentrant, but not for the same cryptoKeyId	
Parameters (in):	cryptoKeyIdCrypto_RandomSeed.cryptoKeyId	Holds the identifier of the key for which a new seed shall be generated.
	entropyseedPtrCrypto_RandomSeed.entropyseedPtr	Holds a pointer to the memory location which contains the data to feed the entropy seed.
	entropyseedLengthCrypto_RandomSeed.entropyseedLength	Contains the length of the entropy in bytesseed in bytes.
Parameters (inout):	None	
Parameters (out):	None	
Return value:	Std_ReturnType	E_OK: Request successful E_NOT_OK: Request Failed
Description:	This function generates the internal seed state using the provided entropy source. Furthermore, this function can be used to update the seed state with new entropy	

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77263: [CRYPTO] API function argument naming between Csm, CryIf and Crypto

  Problem description:

  In [SWS_Csm_01051] "Csm_RandomSeed" and [SWS_CryIf_91007] "CryIf_RandomSeed" function arguments are named "seedPtr and "seedLength". In [SWS_Crypto_91013] "Crypto_RandomSeed" the same arguments are named "entropyPtr" and "entropyLength".

  Agreed solution:

  [SWS_Crypto_91013]

  rename argument "entropyPtr to "seedPtr"
  rename argument "entropyLengthto "seedLength"

  Correct argument description as well
  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
–Last change on issue 77263 comment 6–

BW-C-Level:

Application	Specification	Bus
1	1	1

1.104 Specification Item SWS_Crypto_91014

Trace References:

none

Content:

Service name:	Crypto_KeySetValidCrypto_KeySetValid	
Syntax:	Std_ReturnType Crypto_KeySetValid(uint32 cryptoKeyId)	
Service ID[hex]:	0x05	
Sync/Async:	Synchronous	
Reentrancy:	Non Reentrant	
Parameters (in):	cryptoKeyIdCrypto_KeySetValid.cryptoKeyId	Holds the identifier of the key which 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:	Sets the key state of the key identified by cryptoKeyId to valid.	

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #76783: Typo or copy/paste mistakes

  Problem description:

  Hello,

  I found some other mistakes in the specification documents. Most of them are typos or copy/paste mistakes. As document owner of the CryptoServiceManager, I need a confirmation from someone else, before I can implement them into the document.

AUTOSAR_SWS_CryptoDriver:
[SWS_Crypto_00139]: CRYPTO_E_KEY_EXTRACT_DENIED does not exist anymore. Replace error code with CRYPTO_E_KEY_READ_FAIL.
[SWS_Crypto_91005]: Crypto_KeyValidSet() shall be named analogously to Csm_KeySetValid() and CryIf_KeySetValid(). Therefore, rename Crypto_KeyValidSet() to Crypto_KeySetValid().
[SWS_Crypto_00071]: In table: inputLengthPtr, secondaryInputLengthPtr, tertiaryInputLengthPtr are no pointer anymore. rename them to inputLength, secondaryInputLength, tertiaryInputLength

AUTOSAR_SWS_CryptoServiceManager:
[SWS_Csm_01035]: Csm_KeyCopy() shall call CryIf_KeyCopy() not CryIf_KeyElementCopy().
[SWS_Csm_01080]: Csm_AsymPrivateKeyType is not up-to-date. It should be modified like [SWS_Csm_00076] Csm_AsymPublicKeyType or [SWS_Csm_01082] Csm_SymKeyType.
SWS_Csm_00455
[SWS_Csm_00455]: tag as obsolete
[ECUC_Csm_00188]: typo: CsmMacGenerateAlgorithmFamiliy -> CsmMacGenerateAlgorithmFamily
[ECUC_Csm_00049]: CsmMacVerifyAlgorithmMode missing. (see analogues CsmMacGenerateAlgorithmMode [ECUC_Csm_00189])
[SWS_Csm_00966]: CopyPaste mistake: Delete: "Wrong return values - here are the correct ones:"
[SWS_Csm_01023]: plaintextLength description wrong. replace with "Contains the number of bytes to encrypt."
[SWS_Csm_01023]: typo "associtatedDataLengthPtr" and it is no pointer. replace with: "associatedDataLength"
[SWS_Csm_01025]: typo, replace line "job->jobPrimitiveInputOutput.outputLength = ciphertextLength," with
"job->jobPrimitiveInputOutput.outputLengthPtr = ciphertextLengthPtr,"
[SWS_Csm_01013]: typo: rename "PrimitiveInputOutput" to "jobPrimitiveInputOutput". Or is this rename intended?
Then every assignment of "jobPrimitiveInputOutput" has to be renamed to "primitiveInputOutput" like
"job->jobPrimitiveInputOutput.mode = mode," has to be modified to "job->primitiveInputOutput.mode = mode,"
[SWS_Csm_01026]: typo: replace "associtatedDataLength" with "associatedDataL-

ength"
[SWS_Csm_01027]: missing line: "job->jobPrimitiveInputOutput.verifyPtr = verifyPtr."
[SWS_Csm_00992]: copypaste mistake: replace "mode: The Crypto_JobInfoType job with the corresponding jobId shall be modified in the following way:" with ""mode: Indicates which operation mode(s) to perfom."
[SWS_Csm_00992]: copypaste mistake: replace "resultLengthPtr: Contains the number of bytes of the associated data." with ""resultLengthPtr: Holds a pointer to the memory location in which the output length in bytes of the signature is stored. On calling this function, this parameter shall contain the size of the buffer provided by resultPtr. When the request has finished, the actual length of the returned value shall be stored."
[SWS_Csm_01543]: description wrong. replace with "Generate a random number and stores it in the memory location pointed by the result pointer."
[SWS_Csm_00168]: description wrong, there is no IV. replace with "This function is deprecated. Sets the key for symmetrical encryption."
[SWS_Csm_01031]: description wrong, it is not decrement. "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterRead Service"

A proposed solution is added, too.

Agreed solution:

AUTOSAR_SWS_CryptoDriver:
[SWS_Crypto_00139]: Replace CRYPTO_E_KEY_EXTRACT_DENIED with CRYPTO_E_KEY_READ_FAIL.
[SWS_CryIf_91015]: Remove CRYPTO_E_KEY_EXTRACT_DENIED
[SWS_Crypto_91005]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add Crypto_KeySetValid as API (Description according to SWS_Crypto_91005)
[SWS_Crypto_00082]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add E_UNINIT DET check SWS for Crypto_KeySetValid (Text according to SWS_Crypto_00082)
[SWS_Crypto_00083]: Set Crypto_KeyValidSet obsolete.
[SWS_Crypto_xxx]: Add E_PARAM_HANDLE DET check SWS for Crypto_KeySetValid (Text according to SWS_Crypto_00083)
last sentence in 8.2.4.1.2: Rename Crypto_KeyValidSet to Crypto_KeySetValid

[SWS_Crypto_00071]: rename inputLengthPtr, secondaryInputLengthPtr, tertiaryInputLengthPtr to inputLength, secondaryInputLength, tertiaryInputLength

AUTOSAR_SWS_CryptoServiceManager:
[SWS_Csm_01035]: CryIf_KeyElementCopy() shall be replaced with CryIf_KeyCopy().

[SWS_Csm_01080]: replace with (see [SWS_Csm_00076]):
Name: Csm_AsymPrivateKeyType
Kind: Structure
Elements:
length: uint32: This element contains the length in bytes of the key stored in element 'data'
data: Csm_AsymPrivateKeyArrayType: This element contains the key data or a key handle.
Description: Structure for the private asymmetrical key.
Variation: –

[SWS_Csm_00455]: tag as obsolete

[ECUC_Csm_00188]: typo: CsmMacGenerateAlgorithmFamiliy -> CsmMacGenerateAlgorithmFamily
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmMode (see analogues CsmMacGenerateAlgorithmMode [ECUC_Csm_00189])
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmModeCustom (see analogues CsmMacGenerateAlgorithmModeCustom [ECUC_Csm_00189])
[ECUC_Csm_00049]: add CsmMacVerifyAlgorithmKeyLength (see analogues CsmMacGenerateAlgorithmKeyLength [ECUC_Csm_00189])
[SWS_Csm_00966]: Delete: "Wrong return values - here are the correct ones:"
[SWS_Csm_01023]: Replace description with: "Contains the number of bytes to encrypt."
[SWS_Csm_01023]: Replace "associatedDataLengthPtr" with "associatedDataLength"
[SWS_Csm_01025]: Replace line "job->jobPrimitiveInputOutput.outputLength = ciphertextLength," with
"job->jobPrimitiveInputOutput.outputLengthPtr = ciphertextLengthPtr,"
[SWS_Csm_01013]: rename "PrimitiveInputOutput" to "jobPrimitiveInputOutput". rename "state" to "jobState".
[SWS_Csm_01026]: replace "associtatedDataLength" with "associatedDataLength"
[SWS_Csm_01027]: add line: "job->jobPrimitiveInputOutput.verifyPtr = verifyPtr."
[SWS_Csm_00992]: replace "mode: The Crypto_JobInfoType job with the corresponding jobId shall be modified in the following way:" with ""mode: Indicates which operation mode(s) to perform."
[SWS_Csm_00992]: replace "resultLengthPtr: Contains the number of bytes of the associated data." with ""resultLengthPtr: Holds a pointer to the memory location in which the output length in bytes of the signature is stored. On calling this function, this parameter shall contain the size of the buffer provided by resultPtr. When the request has finished, the actual length of the returned value shall be stored."
[SWS_Csm_01543]: replace description with "Generate a random number and

stores it in the memory location pointed by the result pointer."
[SWS_Csm_00168]: replace description with "This function is deprecated. Sets the key for symmetrical encryption."
[SWS_Csm_01031]: replace "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterDecrement Service" with "CRYPTO_SECCOUNTERREAD 0x0A SecureCounterRead Service"
–Last change on issue 76783 comment 29–

BW-C-Level:

Application	Specification	Bus
4	3	1