Document Title	Utilization of Crypto Services
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	602
Document Classification	Auxiliary

Document Version	1.0.0
Document Status	Final
Part of Release	4.1
Revision	1

Document Change History

18.01.2013	1.0.0	AUTOSAR Administration	Initial release

Disclaimer

This specification and the material contained in it, as released by AUTOSAR, is for the purpose of information only. AUTOSAR and the companies that have contributed to it shall not be liable for any use of the specification.

The material contained in this specification is protected by copyright and other types of Intellectual Property Rights. The commercial exploitation of the material contained in this specification requires a license to such Intellectual Property Rights.

This specification may be utilized or reproduced without any modification, in any form or by any means, for informational purposes only.
For any other purpose, no part of the specification may be utilized or reproduced, in any form or by any means, without permission in writing from the publisher.

The AUTOSAR specifications have been developed for automotive applications only. They have neither been developed, nor tested for non-automotive applications.

The word AUTOSAR and the AUTOSAR logo are registered trademarks.

Advice for users

AUTOSAR specifications may contain exemplary items (exemplary reference models, "use cases", and/or references to exemplary technical solutions, devices, processes or software).

Any such exemplary items are contained in the specifications for illustration purposes only, and they themselves are not part of the AUTOSAR Standard. Neither their presence in such specifications, nor any later documentation of AUTOSAR conformance of products actually implementing such exemplary items, imply that intellectual property rights covering such exemplary items are licensed under the same rules as applicable to the AUTOSAR Standard.

Table of Contents

1 Introduction

1.1 Purpose of the document

This document describes the intended usage of the AUTOSAR specified crypto functionality. The purpose of the document is to guide the user/integrator in application of the AUTOSAR supported, cryptographic functionality.

1.2 Scope of the document

The scope of this document is to help CSM/CAL implementers and integrators to understand:

- architecture of AUTOSARs cryptography related software
- integral representation of components and their relationships
- integration with cryptography supporting hardware

2 Acronyms and abbreviations

Acronym:	***Description:***
CSM	Crypto Service Manager
CAL	Crypto Abstraction Library
CPL	Cryptographic Primitives Library

Abbreviation:	***Description:***
CRY	Cryptographic Primitives Module

Terminology:	***Description:***

3 Related documentation

3.1 Input documents

[1] Requirements on Crypto Service Manager
AUTOSAR_SRS_CryptoServiceManager.pdf

[2] Requirements on Libraries
AUTOSAR_SRS_Libraries.pdf

[3] Specification of Crypto Abstraction Library
AUTOSAR_SWS_CryptoAbstractionLibrary.pdf

[4] Specification of Crypto Service Manager
AUTOSAR_SWS_CryptoServiceManager.pdf

[5] Layered Software Architecture
AUTOSAR_EXP_LayeredSoftwareArchitecture.pdf

[6] HIS, http://portal.automotive-his.de

3.2 Related standards and norms

None

4 Summary

Starting with AUTOSAR 4.0, cryptographic requirements were introduced in order to support increasing automotive market demands in that field.

AUTOSAR cryptographic requirements are defined by [1]. In order to implement the requirements, Autosar defines two different specification documents [4] (CSM) and [3] (CAL). Because CAL specifies an AUTOSAR library, it follows the related requirements document (see [2]) as well.

According to [5] an AUTOSAR Service Manager, like the CSM, represents a basic software component which controls the concurrent, multiple and synchronous/asynchronous access of one or multiple clients to one or more services. I.e. it performs buffering, queuing, arbitration**,** multiplexing. In general, such managers are located in the Services Layer.

According to [5] an AUTOSAR Library like CAL represents a stateless basic software component that provides functionality to SW-Cs and BSW modules (e.g. CDDs).

According to [5] hardware access is forbidden for an AUTOSAR library. Therefore, for AUTOSAR compliant access on cryptographic hardware, the CSM must use a specific driver. AUTOSAR doesn’t specify such access; any driver of this kind must be provided as a CDD. Some cryptographic hardware specification is accessible at [6].

The AUTOSAR CSM and CAL specifications define the same cryptographic functionality. This provided functionality covers the following areas:

- Hash calculation
- Generation and verification of message authentication codes
- Random number generation
- Encryption and decryption using symmetrical algorithms
- Encryption and decryption using asymmetrical algorithms
- Generation and verification of digital signature
- Key management operations

The AUTOSAR CSM and CAL specifications do not define the cryptographic algorithms to be used in order to implement the required functionality. It is up to the implementer to decide regarding based on his assumptions or (better) based on customer requirements that are not provided by AUTOSAR.

The existence of two BSW modules (a service module and a library) providing the same (or similar) functionality is for historical reasons:

The cryptographic library has been introduced for using cryptographic functionality directly by bypassing the RTE, after the CSM has been defined already. Even if both modules provide mostly the same functionality, both use different mechanisms for communication.

5 Specification elements

5.1 Crypto Service Manager (CSM)

5.1.1 Summary

The CSM is located in the Services Layer of Autosar layered architecture. It offers standardized access to cryptographic services for applications via the port mechanism.

Cryptographic services are, e.g., the computation of hashes, the verification of asymmetrical signatures, or the symmetrical encryption of data. These services depend on underlying cryptographic primitives and cryptographic schemes (CRY). CSM services use cryptographic algorithms that are implemented using a software library or cryptographic hardware modules.

The CSM services are generic. Consequently the CSM allows different applications to use the same service for different algorithms. This is possible due to configurable allocation of so called cryptographic service primitives to the services offered by the CSM:

> E.g., one application might need to use the hash service to compute an MD5 digest and another might need to compute an SHA1 digest.
>
> Or one application might need to verify a signature which has been computed with the RSASSA-PKCS1-V1_5 signature scheme using SHA1 as an underlying hash primitive, while another application might need to verify a signature computed with a different scheme which uses MD5 as an underlying hash primitive.

While the definition of the services, offered by the CSM, is part of the CSM specification document, there is no specification in regards of the cryptographic service primitives contained. It is up to a provider to specify, implement and provide such for access by a CSM implementation.

Furthermore, since the computation of many of the cryptographic services is very computation intensive, provisions have been made for scheduling these long computations. The CSM is configurable to use an asynchronous interface where the service requests are placed at the CSM jobs queue by synchronous interface functions and the jobs are processed later in a regularly scheduled main function.

5.1.2 Locality of the CSM

Due to the fact that the CSM is an element of the AUTOSAR Services Layer, the offered service can be used locally only – it is not possible to access the service of the CSM directly via the VFB from a different ECU.

If this is needed, it is up to a provider to specify, implement and provide some proxy for access of a CSM implementation, utilizing AUTOSAR communication services for instance.

If the CSM is used remotely (via a proxy), it must be taken into account that this raises security implications: any communication between ECUs is done via not protected (somehow open accessible) communication busses (e.g. CAN). This means, that unencrypted date, not yet signed data or key material would be transmitted and might become stolen or manipulated.

5.1.3 Support of Cryptographic Hardware

CSM services use cryptographic algorithms that are implemented using a software library or cryptographic hardware modules – both are out of scope and not specified by AUTOSAR.

Cryptographic hardware module(s) can be placed on or off the MCU.

Any cryptographic hardware module that is placed off MCU must be accessed using SPI for instance. But a system designer must consider that such architecture might be susceptible for threats like 'man in the middle' attacks.

To manage a cryptographic hardware module a driver must be used. Autosar standard do not include this type of driver, so it must be developed using a specific approach and additionally; it is up to a provider to specify, implement and provide such for access by a CSM implementation.

5.1.4 Reentrancy

The CSM supports processing of a single instance of each service at a time and allow parallel access to different services.

5.1.5 Role of the Main Function

The CSM provides a main function only, if asynchronous behavior of the CSM has been configured (see configuration parameter CsmUseSyncJobProcessing).

5.1.5.1 Synchronous CSM operation

All CSM services will be executed immediately and in context of the caller.
In the specific case of core crossing calls (in a multi core environment) any CSM service is executed in context of the RTE and my face some delay.
No Main Function is provided (and scheduled).

5.1.5.2 Asynchronous CSM operation

If asynchronous behavior has been configured, all cryptographic services will be executed in context of its scheduled Main Function.
A user notification must be executed as soon a cryptographic service has been completed (successfully or not). The CSM implementation therefore uses callouts that are specific for each provided cryptographic service (see SWS_Csm_0074).

The CRY primitives use CSM callbacks for notifications (see SWS_Csm_0455 and SWS_Csm_0457).

5.1.6 Configurability

The CSM must be configured before it can be used at all. Especially it is necessary that the CRY is configured, because the CSM API needs to access the CRY primitives for service provision.
This CSM specific configuration comprises the names of the CRY primitives and the maximum size of keys if provided/used by a CRY primitive.

5.2 Crypto Abstraction Library (CAL)

5.2.1 Summary

The CAL provides other software modules with cryptographic services. The CAL offers C functions that can be called directly, i.e. from BSW modules, from SW-C or from Complex Drivers. These functions depend on underlying cryptographic primitives and cryptographic schemes (CPL).

As the CAL is a pure library, it is not related to a special layer of the AUTOSAR Layered Software Architecture. The services of the CAL are always executed in the context of the calling function.

Each service is accessed using reentrant and synchronous functions. To implement reentrancy all functions have as parameter a context buffer that is supplied by the caller and contains the entire status of a service needed to start or continue a cryptographic operation. By using that parameter, it doesn’t need to store the service status/context in a global area as CSM does.

5.2.2 Support of Cryptographic Hardware

AUTOSAR libraries must be fully reentrant (stateless); because current cryptographic hardware doesn’t support this, a CAL cannot use cryptographic hardware.

5.2.3 Reentrancy

The CAL is fully reentrant; each service can be used by several processes in parallel and concurrently.

5.2.4 Configurability

Contrary to other AUTOSAR Libraries, the CAL must be configured before it can be used at all. Especially it is necessary that the CPL is configured, because CAL API must access the CPL primitives for service provision.

Usually this configuration comprises the names of the CPL primitives and the maximum size of keys if provided/used by a CPL primitive.

5.3 CRY and CPL

CRY and CPL are specified by CSM respectively CAL specification documents. They provide implementations of cryptographic primitives that are used by CSM respectively CAL. The CSM and CAL specifications define the signature of the CRY/CPL APIs in a generic manner but no functionality.

This architecture (separation of interface layer from implementation layer: CSM/CRY and CAL/CPL) allows certain level of abstraction:

- CSM and CAL are generic, because they are independent from any implementation specifics of cryptographic algorithms.
- CRY and CPL provide the capability of different and concurrent implementations of the same cryptographic services.
  E.g. it is possible to implement different hash algorithms and to use them via the same API.
- CSM and CAL provide an input parameter for selection of the cryptographic primitive to be used (e.g. the hash algorithm).

Variants of CRY/CPL modules with different optimization objectives may exist. These variants should be organized in separate modules. Optimizations may include execution speed, platform specific optimizations, RAM size and/or code segment size etc. The most suitable variant for a given deployment should be used.

Many CRY/CPL interfaces use the same cryptographic building blocks. Thus, cryptographic building blocks should be implemented as separate modules and be called from the CRY/CPL interfaces. This implies that the code for cryptographic building blocks should not be implemented more than once.

6 CSM and CAL provided Cryptographic Services

This chapter contains a non normative list of interfaces that are provided by the Crypto Service Manager respectively by the Crypto Abstraction Library.

6.1 Services for Cryptography using symmetric Keys

These services utilize symmetric keys. Even if those services are used for handling of asymmetric keys, they are collected there.

Service	CSM name	CAL name	Streaming API
Message Authentication Code	Csm_MacGenerate Csm_MacVerify	Cal_MacGenerate Cal_MacVerify	Yes
Block Cipher	Csm_SymBlockEncrypt Csm_SymBlockDecrypt	Cal_SymBlockEncrypt Cal_SymBlockDecrypt	Yes
Stream Cipher	Csm_SymEncrypt Csm_SymDecrypt	Cal_SymEncrypt Cal_SymDecrypt	Yes
Key Generation	Csm_SymKeyGenerate	N/A	No
	Csm_KeyDerive	Cal_KeyDerive	Yes
	Csm_KeyDeriveSymKey	N/A	No
Key Update	Csm_SymKeyUpdate	N/A	Yes
Key Extraction	Csm_SymKeyExtract	Cal_SymKeyExtract	Yes
Key Wrapping	Csm_SymKeyWrapSym	Cal_SymKeyWrapSym	Yes
Wrapping asymmetric private Key using symmetric Key	Csm_AsymPrivateKeyWrapSym	Cal_AsymPrivateKeyWrapSym	Yes

6.2 Services for Cryptography using asymmetric Keys

These services utilize asymmetric keys.

Service	CSM name	CAL name	Streaming API
Cipher using Public Key	Csm_AsymEncrypt Csm_AsymDecrypt	Cal_AsymEncrypt Cal_AsymDecrypt	Yes
Signature	Csm_SignatureGenerate Csm_SignatureVerify	Cal_SignatureGenerate Cal_SignatureVerify	Yes
Key Exchange	Csm_KeyExchangeCalcPubVal	Cal_KeyExchangeCalcPubVal	No
	Csm_KeyExchangeCalcSecret	Cal_KeyExchangeCalcSecret	Yes
	Csm_KeyExchangeCalcSymKey	N/A	Yes
Wrapping symmetric Key using public Key	Csm_SymKeyWrapAsym	Cal_SymKeyWrapAsym	Yes
Key Update	Csm_AsymPrivateKeyUpdate Csm_AsymPublicKeyUpdate	N/A	Yes
Key	Csm_AsymPublicKeyExtract	Cal_AsymPublicKeyExtract	Yes

Extraction	Csm_AsymPrivateKeyExtract	Cal_AsymPrivateKeyExtract	
Wrapping asymmetric private Key using public Key	Csm_AsymPrivateKeyWrapAsym	Cal_AsymPrivateKeyWrapAsym	Yes

6.3 Other Cryptographic Services

These services neither utilize symmetric nor asymmetric keys.

Service	CSM name	CAL name	Streaming API
Hash calculation	Csm_Hash	Cal_Hash	Yes
Random	Csm_RandomSeed	Cal_RandomSeed	Yes
	Csm_RandomGenerate	Cal_RandomGenerate	No
Compression	Csm_Compress Csm_Decompress	Cal_Compress Cal_Decompress	Yes
Checksum	Csm_Checksum	Cal_Checksum	Yes