Document Title	Specification of Identity and Access Management
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	900

Document Status	Final
Part of AUTOSAR Standard	Adaptive Platform
Part of Standard Release	18-10

Document Change History			
Date	**Release**	**Changed by**	**Description**
2018-10-31	18-10	AUTOSAR Release Management	• Reworked functional specification • Removed API specification for general rework
2018-03-29	18-03	AUTOSAR Release Management	• Initial release

Disclaimer

Table of Contents

1 Introduction and functional overview

This specification describes the functionality, API and the configuration for the Identity and Access Management functional cluster of the AUTOSAR Adaptive Platform.
The Identity and Access Management offers applications a standardized interface to access management operations.

2 Acronyms and Abbreviations

The glossary below includes acronyms and abbreviations relevant to the Identity and Access Management module that are not included in the AUTOSAR glossary [1].

Abbreviation / Acronym:	Description:
PDP	Policy Decision Point
PEP	Policy Enforcement Point
IPC	Inter-Process Communication

3 Related documentation

3.1 Input documents & related standards and norms

[1] Glossary
AUTOSAR_TR_Glossary

[2] Requirements on Identity and Access Management
AUTOSAR_RS_IdentityAndAccessManagement

3.2 Related standards and norms

See chapter 3.1.

3.3 Related specification

See chapter 3.1.

4 Constraints and assumptions

4.1 Known Limitations

- The topic of providing identity information of Adaptive Applications to PEPs is still under discussion. Requirements and specification details regarding Application ID / Application Instance ID and providing application identity in general may be affected by this discussion and may change accordingly.
- No API since there was no consensus.

4.2 Assumptions

The integrator can configure a secure channel between `Policy Decision Points` and `Policy Enforcement Points`. This could be done through the operating system's access rights for example.

5 Dependencies to other modules

There are currently no dependencies to other functional clusters.

6 Requirements Tracing

The following tables reference the requirements specified in [2] and links to the fulfillment of these. Please note that if column "Satisfied by" is empty for a specific requirement this means that this requirement is not fulfilled by this document.

Requirement	Description	Satisfied by
[RS_IAM_00002]	Enforcement of access control shall happen within `Adaptive Platform Foundation`	[SWS_IAM_01001] [SWS_IAM_02009]
[RS_IAM_00003]	Applications shall be prevented from taking control over the AUTOSAR PEP	[SWS_IAM_01000]
[RS_IAM_00004]	Circumvention of AUTOSAR PEP interfaces by Applications shall be prevented.	[SWS_IAM_02010]
[RS_IAM_00005]	`Adaptive Platform Foundation` shall enforce that only Applications that are configured accordingly are able to gain information about the permissions of other applications	[SWS_IAM_01000] [SWS_IAM_02000]

Requirement	Description	Satisfied by
[RS_IAM_00008]	Access shall be denied by the PEP if the corresponding PDP is not available	[SWS_IAM_02004]
[RS_IAM_00009]	An `Adaptive Application` may provide access control decisions	[SWS_IAM_02001] [SWS_IAM_02005] [SWS_IAM_02006] [SWS_IAM_02007] [SWS_IAM_02008] [SWS_IAM_02011]
[RS_IAM_00010]	Adaptive applications shall only be able to use AUTOSAR Resources when authorized	[SWS_IAM_02001] [SWS_IAM_02003]
[RS_IAM_00011]	Policies shall be enforced by the local `Adaptive Platform Foundation`	[SWS_IAM_02003] [SWS_IAM_02005] [SWS_IAM_02006] [SWS_IAM_02007] [SWS_IAM_02008] [SWS_IAM_02011] [SWS_IAM_02012] [SWS_IAM_02013] [SWS_IAM_02014] [SWS_IAM_02015]
[RS_IAM_00012]	No description	[SWS_IAM_02011] [SWS_IAM_02012] [SWS_IAM_02013] [SWS_IAM_02014] [SWS_IAM_02015]
[RS_IAM_00014]	Unique `Adaptive Application` ID	[SWS_IAM_02013]
[RS_IAM_00015]	No description	[SWS_IAM_02013]

7 Functional specification

The AUTOSAR Adaptive Platform organizes the software of the AUTOSAR Adaptive Foundation as functional clusters. These clusters offer common functionality as services to the applications. The Identity and Access Management (IAM) for AUTOSAR Adaptive Platform is such a functional cluster and is part of “AUTOSAR Runtime for Adaptive Applications” - ARA. The functional cluster may consist of multiple modules. It is responsible for verifying identities and managing access control to resources.

The Identity and Access Management provides the infrastructure to perform access control for intra-ECU and inter-ECU operations. This infrastructure consists of platform components and optionally of OEM-specific applications.

This specification includes the syntax of the API to integrate an OEM-specific application, the relationship of the API to the model and describes the semantics and behavior. The specification does not pose constraints on the internal architecture and implementation.

7.1 Architectural concepts

The Identity and Access Management of AUTOSAR Adaptive Platform can be logically divided into the following components:

- **Policy Enforcement Point (PEP)**
  The PEP is usually implemented in a functional cluster software component and

will query a PDP for allowance to perform an operation and will block the operation if necessary.

- **Policy Decision Point (PDP)**
  The PDP may be implemented in several locations (e.g. an OEM-specific application, an application provided by the stack vendor, within another `Functional Cluster` or as a standalone `Functional Cluster`) and manages access rights.

The related software components can be divided into the following components:

- Language binding
- Optional OEM-specific PDP application
- Identity and Access Management daemon

There are several types of interfaces available in the context of the Identity and Access Management:

- **Public Interface**
  The public interface is specified in this document and part of the ARA namespace (ara::iam) as a standardized API.
- **Protected Interface**
  The protected interface is used for interaction between functional clusters. It could be implemented as a custom API or by re-using the public interface. Note: using the public interface is encouraged.

For the design of the ARA API the following constraints apply:

- Support the independence of application software components from a specific platform implementation
- Make the API as lean as possible, no specific use cases are supported which could also be layered on top of the API

Therefore the API of the Identity and Access Management follows a specific set of design decisions:

- It uses a pure virtual API to integrate different PDP application through a unified interface

7.2 Identity and Access Management Concept and Functional Cluster

The Identity and Access Management `Functional Cluster` is a component in the AUTOSAR Adaptive Platform interacting with other `Functional Clusters` managing resources. These resources may be any of the following non-exhaustive list:

- Cryptographic keys

- Service instances
- Files
- Key-Value-Stores

The purpose of the interaction is to restrict access to these resources based on the modeling within the Application Design and the Deployment Design. This generally requires the following information:

- The identity of the `Application` requesting the operation
- The kind of operation that shall be performed
- The resource(s) on which the operation shall be performed
- A collection of rules describing the allowed operations per resource and `Application`

The identity of an `Application` can be derived from the Deployment Design model since each `Application` is executed in its own `Process`. The identification of a `Process` can be achieved in different ways and generally depends on the used operating system. The following non-exhaustive list provides some suggestions:

- Using individual UIDs per `Application` that can be queried by a `Functional Cluster`
- Using an identification token generated by the `Execution Management` and passed to a `Functional Cluster` during the interaction

The kind of operation that shall be performed is defined by the `Functional Cluster`'s API. Operations may also be clustered into a group of operations depending on the functional specification of the respective `Functional Cluster` (e.g. Modifying operation on cryptographic keys vs. operations using a cryptographic key). The identification of these operations or operation groups is implementation-specific.

The resource an operation shall be performed upon is defined by the `Functional Cluster`'s modeling. All interactions are described by `PortPrototype` in the Application Design model. This `PortPrototype` references a `Functional Cluster`-specific `PortInterface` describing the resource on Application Design level. Each `Functional Cluster`'s Deployment Design model assign the Application Design model's reference a concrete instance of each resource.

The aforementioned information regarding operations and resources can be compiled into a set of rules per `Application` hosted in a `Process`. Therefore, the Identity and Access Management `Functional Cluster` can aggregate all the information. The `Functional Cluster` managing the resource can identify an `Application` at runtime. The interaction between the `Functional Cluster` managing the resource and Identity and Access Management `Functional Cluster` shall include the `Application`'s identity, the operation (group) and the resource. Given this information the Identity and Access Management `Functional Cluster` can grant or deny the access based on its compiled information from the model. Therefore, the `Func-`

tional Cluster managing the resource is the Policy Enforcement Point and the Identity and Access Management Functional Cluster is the Policy Decision Point.

The execution of an Application and a Functional Cluster managing a resource shall be isolated. A possible isolation model is isolation by process separation.

[SWS_IAM_01001] Isolation of an Application from a Functional Cluster ⌈ The Application and the Functional Cluster shall be isolated from each other. ⌋ *(RS_IAM_00002)*

This kind of isolation is however not applicable to all Functional Clusters. In these cases the entire Functional Cluster's implementation is hosted within the Application's process. The Identity and Access Management concept generalizes the idea of the Identity and Access Management functional cluster to apply to the cases. Resources falling into this category may be any of the following non-exhaustive list:

- Files
- Key-Value-Stores

Since the restriction cannot be securely enforced within the Functional Cluster, the underlying operating system's mechanisms should be used. An example for such a solution is employing the aforementioned usage of UIDs per Application and the subsequent configuration of file system permissions for the resource instances. This approach requires implementation-specific extensions of the Execution Management and the Update and Configuration Management.

7.3 Integration of Applications and Identity and Access Management

Any Application may implement a Policy Decision Point. This can be done by implementing the interface PolicyDecisionPoint::Check and registering the implementation using PolicyDecisionPointManagement::RegisterPolicyDecisionPoint. This Application may then also be responsible for exchanging access rights information with remote Adaptive Platform instances and locally managing access rights information. Implementing a separate Application enables the integration of more flexible Policy Decision Points without the need to change the Adaptive Platform implementation. This might be used to update the local access rights configuration after installing an update on a remote ECU.

Using a secure local channel the implemented Policy Enforcement Points can query registered Policy Decision Points. The underlying implementation of the secure local channel is implementation-specific and hidden within PolicyDecisionPointManagement implementation. Thus, the PolicyDecisionPoint implementation shall only implement managing access rules and making policy decisions.

[SWS_IAM_01000] Secure local channel ⌈ A secure local channel must be configured for integrating a `Policy Decision Point` with the implemented `Policy Enforcement Points`. The security of the local channel must be enforced by the operating system. It shall enforce access control such that only authorized `Applications` can register `Policy Decision Points`. ⌋*(RS_IAM_00003, RS_IAM_00005)*

The implementation of `Policy Enforcement Points` is specific to the `AUTOSAR Adaptive Platform` implementation. However the `Policy Enforcement Points` need to format their query according to [SWS_IAM_02011], [SWS_IAM_02012], [SWS_IAM_02013], [SWS_IAM_02014] and [SWS_IAM_02015].

A Not applicable requirements

All mentioned requirements are applicable.