authorization-control-intra

Service Description

Abstract
This document provides service description for the **authorization-control-intra** service.

Contents

1 Overview

This document describes the **authorization-control-intra** service, which enables authorization control within a local cloud. The purpose of this service is to grant access right for a consumer to a provider-service-interface triplet.

The rest of this document is organized as follows. In Section 2, we describe the abstract message functions provided by the service. In Section 3, we end the document by presenting the data types used by the mentioned functions.

1.1 How This Service Is Meant to Be Used

Primarily the Orchestrator Core System should consume this service during the orchestration process to check whether the specified consumer system has right to consume the actually matching provider-service-interface triplet.

1.2 Important Delimitations

The intra-cloud authorization rules are possible to define by database record ids, so when a system or service definition or iterface name has been unregistered or removed, than the authorization rule is going to be remvoed as well.

1.3 Access policy

This service is available only for the Orchestrator and the Choreographer Core Systems.

2 Service Interface

This section describes the interfaces to the service. The **authorization-control-intra** service is used to verify authorization rules. The various parameters are representing the necessary system and service input information. In particular, each subsection names an interface, an input type and an output type, in that order. The input type is named inside parentheses, while the output type is preceded by a colon. Input and output types are only denoted when accepted or returned, respectively, by the interface in question. All abstract data types named in this section are defined in Section 3.

The following interfaces are available.

2.1 interface HTTP/TLS/JSON (CheckAuthRuleRequest) : CheckAuthRuleResponse

Profile ype	Type	Version
Transfer protocol	HTTP	1.1
Data encryption	TLS	1.3
Encoding	JSON	RFC 8259 [1]
Compression	N/A	-

Table 1: HTTP/TLS/JSON communication details.

3 Information Model

Here, all data objects that can be part of the **authorization-control-intra** service provides to the hosting System are listed in alphabetic order. Note that each subsection, which describes one type of object, begins with the *struct* keyword, which is used to denote a collection of named fields, each with its own data type. As a complement to the explicitly defined types in this section, there is also a list of implicit primitive types in Section 3.6, which are used to represent things like hashes and identifiers.

3.1 struct CheckAuthRuleRequest

Field	Type	Mandatory	Description
consumer	SystemDescriptor	yes	Descriptor of the consumer system.
providerInterfaceIds	List<ProviderInterfaceIds>	yes	Array of provider and interface reference objects
serviceDefinitionId	Number	yes	Identifier of the service definition database record.

3.2 struct SystemDescriptor

Field	Type	Mandatory	Description
address	Address	yes	Network address.
authenticationInfo	String	no	Public key of the client certificate.
metadata	Metadata	no	Metadata
port	PortNumber	yes	Port of the system.
systemName	Name	yes	Name of the system.

3.3 struct ProviderInterfaceIds

Field	Type	Mandatory	Description
id	Number	yes	Database record identifier of the provider system
idList	List<Number>	yes	List of interface database record identifiers.

3.4 struct CheckAuthRuleResponse

Field	Type	Description
authorizedProviderInterfaceIds	List<ProviderInterfaceIds>	Array of the authorized provider and interface reference objects
consumer	SystemDescriptor	Descriptor of the consumer system.
serviceDefinitionId	Number	Identifier of the service definition database record.

3.5 struct Metadata

An Object which maps String key-value pairs.

3.6 Primitives

Types and structures mentioned throughout this document that are assumed to be available to implementations of this service. The concrete interpretations of each of these types and structures must be provided by any IDD document claiming to implement this service.

Type	Description
Address	A string representation of the address
Object	Set of primitives and possible further objects.
List<A>	An *array* of a known number of items, each having type A.
Name	A string identifier that is intended to be both human and machine-readable.
Number	Decimal number
PortNumber	A Number between 0 and 65535.
String	A chain of characters.

4 References

[1] T. Bray, “The JavaScript Object Notation (JSON) Data Interchange Format,” RFC 8259, Dec. 2017. [Online]. Available: https://rfc-editor.org/rfc/rfc8259.txt

5 Revision History

5.1 Amendments

No.	Date	Version	Subject of Amendments	Author
1	YYYY-MM-DD	4.6.0		Xxx Yyy

5.2 Quality Assurance

No.	Date	Version	Approved by
1	YYYY-MM-DD	4.6.0	