authorization

Service Description

Abstract
This document provides service description for the **authorization** service.

Contents

1 Overview

This document describes the **authorization** service, which enables service consumption permission validations for both providers and consumers. Additionally, providers can lookup, grant and revoke those permissions. An example of this interaction when a provider system creates authorization policies about its offered service. An other example when a consumer can check whether a service is allowed to use before trying an actual service consumption. Event notification permission is also handled by this service in an event publisher/subscriber scenario. To enable other systems to use, to consume it, this service needs to be offered through the ServiceRegistry.

The **authorization** service contains the following operations:

- *grant* creates a provider-level authorization policy;
- *revoke* removes such policy;
- *lookup* lists the provider-owned authorization policies that match the filtering requirements;
- *verify* checks whether a consumer can use a provider's service/service operation or a subscriber can be notified when a publisher publishes a type of event.

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

1.1 How This Service Is Meant to Be Used

Provider systems should call the *grant* operation to create authorization rules about its offered services. If the provider does not provide a service longer, it should use *revoke* operation to remove the related policies. Rules can be changed by *revoking* the existing ones and create the new ones using the *grant* operation.

Consumer system should call the *verify* to check whether a service instance (or a particular operation of that service instance) can be used before attempting the service consumption. Similarly, a provider system should use *verify* when receive a request to decide whether the request should be fulfilled or not.

1.2 Important Delimitations

The requester has to identify itself to use any of the operations.

1.3 Access policy

Available for anyone within the local cloud.

2 Service Operations

This section describes the abstract signatures of each operations of the service. In particular, each subsection names an operation, an input type and one or two output types (unsuccessful operations can return different structure), in that order. The input type is named inside parentheses, while the output type is preceded by a colon. If the operation has two output types, they are separated by a slash. Input and output types are only denoted when accepted or returned, respectively, by the operation in question. All abstract data types named in this section are defined in Section 3.

2.1 operation grant (AuthorizationGrantRequest) : AuthorizationPolicyResponse / ErrorResponse

Operation *grant* creates a provider-level authorization policy. The grant data must meet the following criteria:

- With this operation the requester can only define authorization policy for its own services/event types. If a management-level policy for the same service instance/event type exists, this one will be ignored.
- Target type can be service definition or event type.
- Target is mandatory. Whether it is a service definition name or an event type name, it is case sensitive and must follow the camelCase naming convention. Target can contain maximum 63 characters of letters (english alphabet) and numbers, and have to start with a letter.
- Cloud is a valid cloud identifier which contains a name part and an organization part delimited with an implementation specific delimiter. Both parts are case sensitive, must follow the PascalCase naming convention, can contain maximum 63 characters of letters (english alphabet) and numbers, and have to start with a letter.
- Cloud can be omitted if the policy is about the consumers of the Local Cloud.
- The default policy is mandatory and describes who can use the target when a more specialized policy is not available. In case of event types, only the default policy is allowed to specify.
- Scoped policies is optional and can contain the specialized policies. Scope is a valid operation name. Operation names are case sensitive, must follow the kebab-case naming convention, can contain maximum 63 character of lowercase letters (english alphabet), numbers and dash (-), have to start with a letter, and cannot end with a dash.
- Policies have types that describe how the policies are defined:
  - *Public in cloud*: All consumers of the specified cloud can use the specified target/scope.
  - *Whitelist-based*: All specified consumers (in a list) of the specified cloud can use the specified target/scope.
  - *Blacklist-based*: All consumers of the specified cloud but the specified ones (in a list) can use the specified target/scope.
  - *System-level metadata-based*: Consumers of the specified cloud with a matching system-level metadata can use the specified target/scope.
- Policies with type *Whitelist-based* or *Blacklist-based* has a mandatory system name list parameter. System names are case sensitive, must following the PascalCase naming convention, can contain maximum 63 character of letters (english alphabet) and numbers, and have to start with a letter.
- Evaluating policies with type *System-level metadata-based* requires an online ServiceRegistry.

2.2 operation revoke (AuthorizationRevokeRequest) : OperationStatus / ErrorResponse

Operation *revoke* removes an authorization policy instance from the Local Cloud. The input operation data must meet the following criteria:

- The requester can only remove its own authorization policies.

2.3 operation lookup (AuthorizationLookupRequest) : AuthorizationPolicyListResponse / ErrorResponse

Operation *lookup* lists the provider-owned authorization policies that match the filtering requirements. The lookup data must meet the following criteria:

- With this operation a requester system can only list authorization policies that created by the requester system.
- If a filter expects a list, there is an OR relation between the elements of the filter.
- There is an AND relation between different kind of filters.
- To use this operation, an application system must specify at least one policy instance id OR one target name OR one cloud identifier.
- If target name is specified then target type is mandatory.

2.4 operation verify (AuthorizationVerifyRequest) : AuthorizationVerifyResponse / ErrorResponse

Operation *verify* checks whether a consumer can use a provider's service/service operation or a subscriber can be notified when a publisher publishes a type of event. The input data must meet the following criteria:

- Only the provider OR a consumer can be the requester of this operation, which means that the appropriate field is optional (the other one is mandatory), however if both provider and consumer is specified, one of them must match with the requester's name.
- Provider and consumer field can contain a system name. System names are case sensitive, must following the PascalCase naming convention, can contain maximum 63 character of letters (english alphabet) and numbers, and have to start with a letter.
- Cloud is a valid cloud identifier which contains a name part and an organization part delimited with an implementation specific delimiter. Both parts are case sensitive, must follow the PascalCase naming convention, can contain maximum 63 characters of letters (english alphabet) and numbers, and have to start with a letter.
- Cloud can be omitted if the policy is about the consumers of the Local Cloud.
- Target type can be service definition or event type.

- Target is mandatory. Whether it is a service definition name or an event type name, it is case sensitive and must follow the camelCase naming convention. Targets can contain maximum 63 characters of letters (english alphabet) and numbers, and have to start with a letter.
- Scope is optional and only mattered if target type is service definition. In this case, scope is a valid operation name. Operation names are case sensitive, must follow the kebab-case naming convention, can contain maximum 63 character of lowercase letters (english alphabet), numbers and dash (-), have to start with a letter and cannot ends with a dash.

3 Information Model

Here, all data objects that can be part of the **authorization** service are listed and must be respected by the hosting system. 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 3.16, which are used to represent things like hashes and identifiers.

3.1 struct AuthorizationGrantRequest

Field	Type	Mandatory	Description
authentication	Identity	yes	The requester of the operation.
cloud	CloudIdentifier	no	The cloud of the potential consumers. Omitted in case of the Local Cloud.
targetType	AuthorizationTargetType	yes	The type of the target (service definition or event type).
target	ServiceName / EventTypeName	yes	The target of the rule.
description	String	no	The description of the rule.
defaultPolicy	AuthorizationPolicyRequest	yes	The policy details of the rule which is used when no more specialized policy details are available.
scopedPolicies	ScopedPoliciesRequest	no	A structure that can contain specialized policy details.

3.2 struct Identity

An Object which describes the identity of a system. It also contains whether the identified system has higher level administrative rights.

3.3 struct AuthorizationPolicyRequest

Field	Type	Mandatory	Description
policyType	AuthorizationPolicyType	yes	The type of the policy.
policyList	List<SystemName>	no (yes)	A list of consumer system names. Mandatory in case of list-based policy type.
policyMetadataRequirement	MetadataRequirements	no (yes)	System-level metadata requirements. Mandatory in case of metadata-based policy type.

3.4 struct MetadataRequirements

A special Object which maps String keys to Object, primitive or list values, where

- Keys can be paths (or multi-level keys) which access a specific value in a Metadata structure, where parts of the path are delimited with dot character (e.g. in case of ”key.subkey” path we are looking for the key named ”key” in the metadata, which is associated with an embedded object and in this object we are looking for the key named ”subkey”).
- Values are special Objects with two fields: an operation (e.g. less than) and an actual value (e.g. a number). A metadata is matching a requirement if the specified operation returns true using the metadata value referenced by a key path as first and the actual value as second operands.
- Alternatively, values can be ordinary primitives, lists or Objects. In this case the operation is equals by default.

3.5 struct Metadata

An Object which maps String keys to primitive, Object or list values.

3.6 struct ScopedPoliciesRequest

An Object which maps ServiceOperationName keys to AuthorizationPolicyRequest values.

3.7 struct AuthorizationPolicyResponse

Field	Type	Description
status	OperationStatus	Status of the operation.
instanceId	AuthorizationPolicyInstanceID	Unique identifier of the created policy instance.
authorizationLevel	AuthorizationLevel	Level (provider or management) of the policy.
cloud	CloudIdentifier	The cloud of the potential consumers. In case of the Local Cloud the word LOCAL is used.
provider	SystemName	The name of the system who provides the target of the rule.
targetType	AuthorizationTargetType	The type of the target (service definition or event type).
target	ServiceName / EventType-Name	The target of the rule.
description	String	The description of the rule.
defaultPolicy	AuthorizationPolicyDescriptor	The policy details of the rule which is used when no more specialized policy details are available.
scopedPolicies	ScopedPoliciesDescriptor	A structure that can contain specialized policy details.

Field	Type	Description
createdBy	SystemName	Authorization policy instance was created by this system.
createdAt	DateTime	Authorization policy instance was created at this timestamp.

3.8 struct AuthorizationPolicyDescriptor

Field	Type	Description
policyType	AuthorizationPolicyType	The type of the policy.
policyList	List<SystemName>	A list of consumer system names. Should only be filled in case of list-based policy type.
policyMetadataRequirement	MetadataRequirements	System-level metadata requirements. Should only be filled in case of metadata-based policy type.

3.9 struct ScopedPoliciesDescriptor

An Object which maps ServiceOperationName keys to AuthorizationPolicyDescriptor values.

3.10 struct ErrorResponse

Field	Type	Description
status	OperationStatus	Status of the operation.
errorMessage	String	Description of the error.
errorCode	Number	Numerical code of the error.
type	ErrorType	Type of the error.
origin	String	Origin of the error.

3.11 struct AuthorizationRevokeRequest

Field	Type	Mandatory	Description
authentication	Identity	yes	The requester of the operation.
instanceId	AuthorizationPolicyInstanceID	yes	Unique policy instance id of the rule.

3.12 struct AuthorizationLookupRequest

Field	Type	Mandatory	Description
authentication	Identity	yes	The requester of the operation.
instanceIds	List<AuthorizationPolicyInstanceID>	no (yes)	Requester is looking for policy instances with any of the specified identifiers. Mandatory if no cloudIdentifiers nor targetNames are specified.
cloudIdentifiers	List<CloudIdentifier>	no (yes)	Requester is looking for policy instances that belongs to any of the specified clouds. Mandatory if no instanceIds nor targetNames are specified.
targetNames	List<ServiceName> / List<EventTypeName>	no (yes)	Requester is looking for policy instances that belongs to any of the specified targets (either service definitions or event types). Mandatory if no instanceIds nor cloudIdentifiers are specified.
targetType	AuthorizationTargetType	no (yes)	The type of the specified targets. Mandatory if targetNames are specified.

3.13 struct AuthorizationPolicyListResponse

Field	Type	Description
status	OperationStatus	Status of the operation.
entries	List<AuthorizationPolicyResponse>	List of policy instance results.
count	Number	Number of returned policy instances.

3.14 struct AuthorizationVerifyRequest

Field	Type	Mandatory	Description
authentication	Identity	yes	The requester of the operation.
provider	SystemName	no (yes)	The name of the system that provides the target. Mandatory if the consumer is the requester.
consumer	SystemName	no (yes)	The name of the system that needs access to the target. Mandatory if the provider is the requester.
cloud	CloudIdentifier	no	The cloud of the consumer. Optional, if the consumer is in the Local Cloud.

Field	Type	Mandatory	Description
targetType	AuthorizationTargetType	yes	The type of the target (service definition or event type).
target	ServiceName / EventType-Name	yes	The name of the target.
scope	ServiceOperationName	no	The service operation that the consumer wants to use. Only mattered when the target is a service definition.

3.15 struct AuthorizationVerifyResponse

Field	Type	Description
status	OperationStatus	Status of the operation.
verified	Boolean	The result of the verification.

3.16 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
AuthorizationLevel	String identifiers that specifies whether a rule is created by a provider for its service instances/event types (provider-level) or a higher entity does that (management-level).
AuthorizationPolicyInstanceID	A composite string identifier that is intended to be both human and machine-readable. It consists of the instance's level (provider or management), cloud identifier, provider name, target type and target, each separated by a special delimiter character. Each part must follow its related naming convention.
AuthorizationPolicyType	String identifiers of the various policy types: for whitelist-based policy, for blacklist-base policy, for cloud-level policy and for system-level metadata-based policy.
AuthorizationTargetType	String identifiers that specifies whether a rule is about a service instance or an event type.
Boolean	One out of true or false.
CloudIdentifier	A composite string identifier that is intended to be both human and machine-readable. It consists of the cloud name and the organization name that managing the cloud. Each part must follow the PascalCase naming convention.
DateTime	Pinpoints a specific moment in time.
ErrorType	Any suitable type chosen by the implementor of service.
EventTypeName	A string identifier that is intended to be both human and machine-readable. Must following camelCase naming convention.
List<A>	An *array* of a known number of items, each having type A.
Number	Decimal number.

Type	Description
Object	Set of primitives and possible further objects.
OperationStatus	Logical, textual or numerical value that indicates whether an operation is a success or a failure. Multiple values can be used for success and error cases to give additional information about the nature of the result.
ServiceName	A string identifier that is intended to be both human and machine-readable. Must following camelCase naming convention.
ServiceOperationName	A string identifier that is intended to be both human and machine-readable. Must following kebab-case naming convention.
String	A chain of characters.
SystemName	A string identifier that is intended to be both human and machine-readable. Must following PascalCase naming convention.

4 References

5 Revision History

5.1 Amendments

No.	Date	Version	Subject of Amendments	Author
1	YYYY-MM-DD	5.0.0		Xxx Yyy

5.2 Quality Assurance

No.	Date	Version	Approved by
1	YYYY-MM-DD	5.0.0	