blacklistDiscovery

Service Description

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

Contents

1 Overview

This document describes the **blacklistDiscovery** service, which enables both application and Core/Support systems to get information about the Local Cloud's centrally managed Blacklist. The possible operations provide lookup for relevant blacklist entries for a system and enable to check if an other system is blacklisted within the LC. To enable other systems to use, to consume it, this service needs to be offered through the ServiceRegistry.

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

- *lookup* returns blacklist entries in force that apply to the requester;
- *check* returns if a system name is on the blacklist.

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

In some use cases, it might be important to get some information about the state of the blacklist. An application system can use the **blacklistDiscovery** service's operations to ask if a system is banned from the Local Cloud, even if the question applies to the requester itself. In the latter case, it is also possible to query the entries that cause the ban.

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. The **blacklistDiscovery** service is used to *lookup* for blacklist entries and to *check* if another system is on the blacklist. 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 lookup (Identity) : BlacklistEntryListResponse / ErrorResponse

Operation *lookup* returns the blacklist entries that are in force and apply to the system that the provided identity belongs to. This is the only operation that is available for a system even if it's blacklisted. The input operation data must meet the following criteria:

- It is not possible with this operation to lookup for other system's entries.
- It is not possible with this operation to lookup for records that are expired or inactive.

2.2 operation check (CheckRequest) : CheckResponse / ErrorResponse

Operation *check* returns if an arbitrary system is on the blacklist. The input operation data must meet the following criteria:

- System names are case sensitive, must follow the PascalCase naming convention and have to be unique within the Local Cloud.
- System names can contain maximum 63 character of letters (english alphabet), and numbers, and have to start with a letter.

3 Information Model

Here, all data objects that can be part of the **blacklistDiscovery** 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 3.7, which are used to represent things like hashes and identifiers.

3.1 struct Identity

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

3.2 struct BlacklistEntryListResponse

Field	Type	Description
status	OperationStatus	Status of the operation.
entries	List<BlacklistEntryResponse>	List of blacklist entry results.
count	Number	The total number of corresponding entries.

3.3 struct BlacklistEntryResponse

Field	Type	Description
systemName	SystemName	Unique identifier of the blacklisted system.
createdBy	SystemName	Unique identifier of the system that created the record.
revokedBy	SystemName	Unique identifier of the system that revoked the record. Only appears if the record was revoked.
createdAt	DateTime	Blacklist record was created at this timestamp.
updatedAt	DateTime	Blacklist record was updated at this timestamp.
reason	String	The system was blacklisted because of this reason.
expiresAt	DateTime	Blacklist record expires at this timestamp. Only appears if the record can expire.
active	Boolean	Indicates if the rule defined by the entry is active. Only false if the rule has been explicitly revoked.

3.4 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.

3.5 struct CheckRequest

Field	Type	Mandatory	Description
authentication	Identity	yes	The requester of the operation.
name	SystemName	yes	Unique identifier of system to check.

3.6 struct CheckResponse

Field	Type	Description
status	OperationStatus	Status of the operation.
isBlacklisted	Boolean	Indicates if the system is on the blacklist or not.

3.7 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
Boolean	One out of true or false.
DateTime	Pinpoints a specific moment in time.
ErrorType	Any suitable type chosen by the implementor of service.
List<A>	An *array* of a known number of items, each having type A.
Number	Decimal number.
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.
String	A chain of characters.
SystemName	A string identifier that is intended to be both human and machine-readable. Must follow 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	