

# query-all

## Service Description

### Abstract

This document provides service description for the **query-all** service.

## Contents

<b>1 Overview</b>	<b>3</b>
1.1 How This Service Is Meant to Be Used . . . . .	4
1.2 Important Delimitations . . . . .	4
1.3 Access policy . . . . .	4
<b>2 Service Interface</b>	<b>5</b>
2.1 interface <a href="#">HTTP/TLS/JSON</a> . . . . .	5
<b>3 Information Model</b>	<b>6</b>
3.1 struct <a href="#">ServiceRegistryList</a> . . . . .	6
3.2 Primitives . . . . .	7
<b>4 References</b>	<b>8</b>
<b>5 Revision History</b>	<b>9</b>
5.1 Amendments . . . . .	9
5.2 Quality Assurance . . . . .	9

## 1 Overview

This document describes the **query-all** service, which enables dedicated core systems to get data about all service registry instance by one request, therefore it is an integral part of the implementation of service discovery requirements in Service Registry Mandatory Core System. Examples of this interaction is a core system that needs the available information about every service instance from Service Registry.

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.



ARROWHEAD

Document title  
**query-all**  
Date  
**2022-10-25**

Version  
**4.4.0**  
Status  
**RELEASE**  
Page  
**4 (9)**

## 1.1 How This Service Is Meant to Be Used

The given core system can consume this without any input data.

## 1.2 Important Delimitations

No delimitations.

## 1.3 Access policy

Available only for the following core systems: *Gatekeeper*, *QoS-Monitor*

## 2 Service Interface

This section describes the interfaces to the service. The **query-all** service is used to get every service instance data. In the following, 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 ()** : **ServiceRegistryList**

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 **query-all** 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 3.2, which are used to represent things like hashes and identifiers.

### 3.1 struct **ServiceRegistryList**

Field	Type	Description
data	Array<Object>	List of service instances
count	Number	Size of the result list.

#### 3.1.1 struct **data**

Field	Type	Description
createdAt	DateTime	Service instance record was created at this UTC timestamp.
endOfValidity	DateTime	Service is available until this UTC timestamp.
id	Number	Identifier of the service instance
interfaces	Array<Object>	List of interfaces the service supports.
provider	Object	Descriptor of the provider system record.
secure	SecureType	Type of security the service uses.
serviceDefinitionResponse	Object	Descriptor of the serviceDefinition record.
serviceUri	URI	URI of the service.
metadata	Metadata	Service metadata
updatedAt	DateTime	Service instance record was modified at this UTC timestamp.
version	Version	Version of the service.

#### 3.1.2 struct **interfaces**

Field	Type	Description
createdAt	DateTime	Interface instance record was created at this UTC timestamp.
id	Number	Identifier of the interface instance
interfaceName	Interface	Specified name of the interface.
updatedAt	DateTime	Interface instance record was modified at this UTC timestamp.

### 3.1.3 struct **provider**

Field	Type	Description
address	String	Network address.
authenticationInfo	String	Public key of the client certificate.
createdAt	DateTime	System instance record was created at this UTC timestamp.
id	Number	Identifier of the system instance
metadata	Metadata	Metadata
port	PortNumber	Port of the system.
systemName	Name	Name of the system.
updatedAt	DateTime	System instance record was modified at this UTC timestamp.

### 3.1.4 struct **serviceDefinition**

Field	Type	Description
createdAt	DateTime	Service definition instance record was created at this UTC timestamp.
id	Number	Identifier of the service definition instance
serviceDefinition	Name	Name of the service definition.
updatedAt	DateTime	Service definition instance record was modified at this UTC timestamp.

## 3.2 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
Boolean	One out of <code>true</code> or <code>false</code> .
DateTime	Pinpoints a specific moment in time.
Object	Set of primitives and possible further objects.
Interface	Any suitable type chosen by the implementor of service
List<A>	An <i>array</i> 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
SecureType	Any suitable type chosen by the implementor of service
Version	Specifies a service version.

## 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>





ARROWHEAD

Document title  
**query-all**  
Date  
**2022-10-25**

Version  
**4.4.0**  
Status  
**RELEASE**  
Page  
**9 (9)**

## 5 Revision History

### 5.1 Amendments

No.	Date	Version	Subject of Amendments	Author
1	YYYY-MM-DD	4.4.0		Xxx Yyy

### 5.2 Quality Assurance

No.	Date	Version	Approved by
1	YYYY-MM-DD	4.4.0	