OrchestrationServiceM2 REST-JSON-TLS

**Abstract**

This document defines interface design of the Orchestration Service based on REST. This is the main Service of the Orchestrator Core System, offered to any System in the Local Cloud.

1. Overview

This document describes the Orchestration Service of G3.2 within the REST interface.

This interface uses HTTP or HTTPS, therefore the related CP is valid.

1. Interfaces

The Orchestration Service is offered through 2 different interfaces. The default interface used by most Systems give options to the requester System to influence the Orchestration process, but also requires a complex request payload with mandatory fields. The 2nd interface is specifically implemented for low-level devices, where a request payload is not needed, and only Orchestration Store based Orchestration process can happen. Table 1 describes these two interfaces.

Table 1 Function description

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function** | **URL subpath** | **Method** | **Input** | **Output** |
| Orchestration | “/orchestration” | POST | **ServiceRequestForm** | 200 OK,  BAD\_PAYLOAD,  DATA\_NOT\_FOUND |
| Orchestration | “/orchestration/{systemGroup}/{systemName}” | GET | **-** | 200 OK,  BAD\_PAYLOAD,  DATA\_NOT\_FOUND |

1. Information Model

As per discussed in the SD document, there are two possible request payloads and one response payload. Here, examples are given for these messages for this interface implementation (REST-JSON-TLS).

# ServiceRegistryEntry

{

"providedService": {

"serviceGroup": "Temperature",

"serviceDefinition": "IndoorTemperature",

"interfaces": ["json"],

"serviceMetadata": {

"unit": "celsius"

}

},

"provider": {

"systemGroup": "TemperatureSensors",

"systemName": "InsecureTemperatureSensor",

"address": "0.0.0.0",

"port": 8454

},

"serviceURI": "temperature",

"version": 1,

"isUDP": false

}

# ServiceQueryForm

{

"service": {

"serviceGroup": "Temperature",

"serviceDefinition": "IndoorTemperature",

"interfaces": ["json"],

"serviceMetadata": {

"entry": [

{

"key": "security",

"value": "token"

},

{

"key": "unit",

"value": "celsius"

}

]

}

},

"pingProviders" : false,

"metadataSearch" : false

}

# ServiceQueryResult

{

"serviceQueryData": [{

"UDP": false,

"providedService": {

"interfaces": [

"json"

],

"serviceDefinition": "IndoorTemperature",

"serviceGroup": "Temperature",

"serviceMetadata": {

"entry": [{

"key": "security",

"value": "token"

},

{

"key": "unit",

"value": "celsius"

}

]

}

},

"provider": {

"address": "127.0.0.1",

"port": 8454,

"systemGroup": "TemperatureSensors",

"systemName": "InsecureTemperatureSensor"

},

"serviceURI": "temperature",

"version": 1

}]

}

1. Revision history

# Amendments

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Date | Version | Subject of Amendments | Author |
| 1 | 2017-09-29 | M2 | Initial | Csaba Hegedűs |

# Quality Assurance

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Date | Version | Approved by |
| 1 |  |  |  |
| 2 |  |  |  |