Document title		Document type
Config HTTP/TLS/JSON		**IDD**
Date		Version
2021-03-29		**1.0**
Author		Status
Jens Eliasson		**DRAFT**
Contact		Page
jens.eliasson@thingwave.eu		**1 (6)**

Config HTTP/TLS/JSON

Interface Design Description

Service ID: *"config"*

Abstract

This document describes the Config service IDD for HTTP/TLS/JSON.

Document title
Config HTTP/TLS/JSON
Date
2021-03-29

Version
1.0
Status
DRAFT
Page
2 (6)

Contents

Document title
Config HTTP/TLS/JSON

Date
2021-03-29

Version
1.0

Status
DRAFT

Page
3 (6)

1   Overview

This document describes the HTTP/TLS/JSON variant of the Config Eclipse Arrowhead service. The Config service is used to store and manage configurations and settings for application systems in a local cloud. The Config service also enables cloud operators (sysop) to create new, delete and list configurations.

    This document exists as a complement to the *Config – Service Description* (Config SD) document. For further details about how this service is meant to be used, please consult that document. The rest of this document describes how to realize the Config service using HTTP [1], TLS [2] and JSON [3], both in terms of its interfaces (Section 2) and its information model (Section 3).

	Document title	Version
	Config HTTP/TLS/JSON	**1.0**
	Date	Status
	2021-03-29	**DRAFT**
		Page
		4 (6)

ARROWHEAD

2   Service Interfaces

This section lists the interfaces that must be exposed by the Config service in alphabetical order. In particular, each subsection first names the HTTP method and path used to call the interface, after which it names an abstract interface from the Config service's SD document, as well as input and output types. All interfaces in this section respond with the HTTP status code `200 OK` if called successfully, unless otherwise is stated.

2.1   GET /configuration/config/$systemName

Interface: **GetConfig**
Output: **ConfigurationResponse**

Called to fetch the active configuration for a specific system. The response is a exemplified in Listing 1.

```
1  GET /configuration/config/examplesystem1 HTTP/1.1
2
3  {
4    "id": 31,
5    "systemName": "examplesystem1",
6    "contentType": "text/plain",
7    "data": "cG9ydD04ODAxCnBhdGg9ImV4YW1wbGUvcGF0aCIKaXA9IjEwLjAuMC4yMyI=",
8    "createdAt": "2021-01-26 22:23:13",
9    "updatedAt": "2021-01-26 22:23:13"
10 }
```

Listing 1: A GetConfig invocation.

Code	Type	Description
200	OK	No error
401	UNAUTHORIZED	No valid authorization
404	NOT FOUND	No valid configuration found
500	INTERNAL SERVER ERROR	Database error etc

Table 1: GetConfig status code responses

3   Information Model

The Config service does not restrict the information model. It only supports JSON. Black-box / binary data configuration files are handled by the RawConfig service.

4 References

[1] R. Fielding and J. Reschke, "Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing," RFC 7230, 2018, RFC Editor. [Online]. Available: https://doi.org/10.17487/RFC7230

[2] E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.3," RFC 8446, 2018, RFC Editor. [Online]. Available: https://doi.org/10.17487/RFC8446

[3] T. Bray, "The JavaScript Object Notation (JSON) Data Interchange Format," RFC 7159, 2014, RFC Editor. [Online]. Available: https://doi.org/10.17487/RFC7159

5 Revision History

5.1 Amendments

No.	Date	Version	Subject of Amendments	Author
1	2021-01-15	0.1	Initial	Jens Eliasson
2	2021-01-25	0.5	Text update	Jens Eliasson
3	2021-02-06	0.8	Data models	Jens Eliasson
4	2021-02-09	0.9	Updated service interfaces	Jens Eliasson
4	2021-02-09	1.0	Final version	Jens Eliasson

5.2 Quality Assurance

No.	Date	Version	Approved by
1			