

interiot

INTEROPERABILITY
OF HETEROGENEOUS
IOT PLATFORMS.

D4.5.

Interoperable IoT Framework API and tools v1.

5th October 2017

INTER-IoT

INTER-IoT aim is to design, implement and test interoperability tools, a framework and a methodology that will allow interoperability among different Internet of Things (IoT) platforms.

Most current existing IoT developments are based on “closed-loop” concepts, focusing on a specific purpose and being isolated from the rest of the world. Integration between heterogeneous elements is usually done at device or network level, and is just limited to data gathering. Our belief is that a multi-layer approach to the integration of different IoT devices, networks, platforms, services and applications will allow a global continuum of data, infrastructures and services. Additionally, a reuse and integration of existing and future IoT systems will be facilitated, enabling the creation of a de facto global ecosystem of interoperable IoT platforms.

In the absence of global IoT standards, INTER-IoT results will allow any company to design and develop new IoT devices or services, leveraging on the existing ecosystem, and bringing them to market quickly.

INTER-IoT has been financed by the Horizon 2020 initiative of the European Commission, contract 687283.

INTER-IoT

Interoperable IoT Framework API and tools v1.

Version: 1.0

Security: Public

October 03, 2017

The INTER-IoT project has been financed by the Horizon 2020 initiative of the European Commission, contract 687283



Disclaimer

This document contains material, which is the copyright of certain INTER-IoT consortium parties, and may not be reproduced or copied without permission.

The information contained in this document is the proprietary confidential information of the INTER-IoT consortium (including the Commission Services) and may not be disclosed except in accordance with the consortium agreement.

The commercial use of any information contained in this document may require a license from the proprietor of that information.

Neither the project consortium as a whole nor a certain party of the consortium warrant that the information contained in this document is capable of use, nor that use of the information is free from risk, and accepts no liability for loss or damage suffered by any person using this information.

The information in this document is subject to change without notice.

Executive Summary

The aim of Deliverable 4.5, entitled “Interoperable IoT Framework API and Tools v1”, is to describe the Initial specification and development of INTER-API and associated tools. The deliverable is the first in the line of two deliverables, as D4.5 will be followed by D4.6 in M30. Thus, it can be considered as the first version of the latter deliverable.

The deliverable reports the relation with INTER-LAYER components and with analysis and design of INTER-FW and INTER-API provided in D4.3 (Interoperable IoT Framework Model and Engine v1). It further provides references to source code repositories and software deployments of INTER-FW and INTER-API. These developments represent technical work carried on in two tasks: T4.4 (Implementation of the IoT Interoperable Framework Engine, M13 – M30) and T4.5 (Design and Implementation of the IoT Interoperable Framework APIs and Tools for Programming and Managing Interoperable IoT Platforms, M13 – M30).

In this version of the deliverable, T4.4 work concentrates on basic functionalities of INTER-FW in order to support the needs of the Pilots and demonstrate the usage of INTER-FW. In the date of the publication of this document, there is a version of INTER-FW Web Console with the fully functional implementation of the following features: FIWARE, universAAL and SEAMS2-WSO2 platforms configuration and management, basic virtual gateway management, topology of the virtual network, functional semantics view and access to API Management.

T4.5 efforts provide deployment of a customised INTER-IoT API Manager. In this first phase, INTER-API provides basic user management and authentication mechanisms, API publish/API store/API usage functionalities and an integrated REST access to INTER-LAYER components. The following deliverable will provide configuration and management APIs for INTER-FW and further extensibility mechanisms for INTER-LAYER.

The provided software has been developed in accordance to the initial description of the layered architecture provided in the Description of the Action. It has considered the requirements gathered in D2.3 (INTER-IoT Requirements and Business Analysis), D4.1 (Initial Reference IoT Platform Meta-Architecture and Meta Data Model), D4.2 (Final Reference IoT Platform Meta-Architecture and Meta Data Model - internal draft) and D4.3 (Interoperable IoT Framework Model and Engine v1).

It is important to note that GIT source code repositories and deployments of INTER-FW and INTER-API represent an integral part of this deliverable. Although the final aim is to publish the source code under Apache 2.0 license, at the time of submission the repositories still have restricted access until all formalities related to the license have been solved. Regarding credentials to access the INTER-FW and INTER-API cloud deployment, they have been made available to the European Commission through the submission portal.

List of Authors

Organisation	Authors	Main organisations' contributions
VPF	Pablo Giménez	Provision and documentation of Platforms UX definitions.
UNICAL	Raffaele Gravina	
PRO	Miguel Llácer, Miguel A. Llorente, Miguel Montesinos	Coordination of task T4.4. Development of INTER-FW, provision of sections 2.1 and 2.2.
TU/e	George Exarchakos	Deliverable internal review.
XLAB	Flavio Fuat, Matevž Markovič	Coordination of task T4.5 and leading the compilation of D4.5. Customisation and deployment of the API Request Manager. Provision of sections 1, 2.2 and 3.2.
SRIPAS	Katarzyna Wasielewska-Michniewska, Paweł Szmeja, Wiesław Pawłowski	Provision and documentation of IPSM API and Semantic Repository definitions.
RINICOM	Eric Carlson	Provision and documentation of N2N API definitions.
UPV	Eneko Olivares, Andreu Belsa, Jara Suárez de Puga	Provision and documentation of AS2AS API definitions.

Change control datasheet

Version	Changes	Chapters	Pages
0.1.0	Created base structure	-	9
0.1.1	Introduction draft	1	10
0.2.0	INTER-FW draft	2.1, 3.1	12
0.2.1	INTER-API draft	2.2., 3.2	14
0.2.2	Added screenshots	2, 3	18
0.3.0	Completed missing parts.	All	20
1.0	Format whole document.	All	20
1.1	Internal review corrections	All	20
1.2	Final Quality check	All	20

Contents

Executive Summary	3
List of Authors	4
Change control datasheet.....	5
Contents.....	6
List of Figures	7
List of Tables.....	7
Acronyms	8
1 Introduction	9
1.1 Relation to INTER-LAYER (WP3)	9
1.2 Relation to Deliverable D4.3 Interoperable IoT Framework Model and Engine v1	10
2 Source code repositories	10
2.1 INTER-FW Web Console	10
2.2 INTER-FW API Manager	12
3 Deployments of demo applications.....	14
3.1 INTER-FW Web Console	14
3.2 INTER-FW API Manager	16

List of Figures

Figure 1. INTER-FW GIT repository, list of commits.....	11
Figure 2. INTER-FW GIT repository, directory structure.....	11
Figure 3. INTER-API GIT repository, list of commits	12
Figure 4. INTER-API GIT repository, directory structure	13
Figure 5. INTER-FW Web Console.....	15
Figure 6 INTER-FW Web Console for Management and Listing of Gateways	15
Figure 7 INTER-FW Web Console for access information on a Gateway	15
Figure 8 INTER-FW Web Console for management and listing of physical devices attached to the Gateway	16
Figure 9 INTER-FW Web Console for accessing the HTTP interface of a Gateway	16
Figure 10. API Manager Store: API Console for testing APIs.....	17
Figure 11. API Manager Store: Active subscriptions for a selected user	18
Figure 12. API Manager Publisher: An example of the API lifecycle management	18

List of Tables

Table 1. List of INTER-API related GIT repositories	12
Table 2. Web Console user names and roles	14
Table 3. API Manager access links.....	17

Acronyms

API	Application Programming Interface
AS2AS	Application Service Layer Interoperability
D2D	Device Layer Interoperability
D3.1	Deliverable 3.1: Methods for Interoperability and Integration v.1
D3.2	Deliverable 3.2: Methods for Interoperability and Integration v.2
D4.3	Deliverable 4.3: Interoperable IoT Framework Model and Engine v1
D4.5	Deliverable 4.5: Interoperable IoT Framework API and Tools v1
D4.6	Deliverable 4.6: Interoperable IoT Framework API and Tools v2
DS2DS	Data and Semantics Layer Interoperability
GUI	Graphical User Interface
INTER-API	INTER-FW and INTER-LAYER exposed as software APIs
INTER-FW	INTER-IoT Interoperable IoT Framework
INTER-LAYER	INTER-IoT Layer integration tools
INTERMW	Middleware Layer Interoperability
IoT	Internet of Things
N2N	Network Layer Interoperability
QoS	Quality of Service
REST	Representational state transfer
URL	Uniform Resource Locator
WP3	INTER-IoT Work Package 3: Layer Interoperability
WP4	INTER-IoT Work Package 4: Interoperability Framework API
WSO2	WSO2 is an open source technology provider

1 Introduction

This deliverable describes the available source code repositories and deployed applications that are a result of implementation of initial versions of INTER-FW and INTER-API. The first section describes the relation with other parts of the INTER-IoT project, namely WP3 that provides INTER-LAYER components and D4.3, which provides the analysis and design of components provided in this deliverable.

1.1 Relation to INTER-LAYER (WP3)

Implementation of INTER-FW and INTER-API is closely related to the outputs of WP3. All WP3 components provide a REST API: Device Layer Interoperability (D2D), Network Layer Interoperability, Middleware Layer Interoperability (INTERMW), Application Service Layer Interoperability (AS2AS) and the Data and Semantics Layer Interoperability (DS2DS). The REST API interface of INTER-LAYER components is provided in the OpenAPI (Swagger) format.

A detailed description of INTER-LAYER and relations with INTER-FW and INTER-API (WP4) can be found in the following documentation:

- **D3.1 Methods for Interoperability and Integration v.1**, which describes the initial approach and architecture of INTER-LAYER.
- **D3.2 Methods for Interoperability and Integration v.2 (forthcoming)**, which describes more in detail design and technical choices of the implementation of INTER-LAYER components and the interfaces they provide.
- **INTER-LAYER REST API definitions git repository:** https://git.INTER-IoT.eu/INTER-IoT/layer_apis¹
- **Deliverable D4.3 Interoperable IoT Framework Model and Engine v1**, which describes:
 - the relation between WP3 and WP4 in the section 3.6 *INTER-FW relation with INTER-LAYER*;
 - A summary of INTER-LAYER REST APIs in the section 4.3.1. *INTER-LAYER REST API Endpoints*.

¹ Access to the different applications may need a password that may be provided upon request

1.2 Relation to Deliverable D4.3 Interoperable IoT Framework Model and Engine v1

This deliverable is closely related to D4.3 in the sense that it provides the implementation of the first phase (v1) of INTER-FW and INTER-API components specified there.

The following components have been implemented according to the analysis and design provided in D4.3:

- INTER-FW Web application that consists of the following
 - Back-end: D4.3 chapter 2.3 *Back-end analysis* and chapter 3.3 *INTER-FW Components*.
 - Front-end: D4.3 chapter 2.4 *Front-end analysis* and chapter 3.5 *INTER-FW GUI design*.
- INTER-API, which provides an unified REST API access to INTER-LAYER:
 - Analysis is provided in the D4.3 chapter 2.5 *API analysis* and the design in chapter 4 *INTER-API*.

2 Source code repositories

2.1 INTER-FW Web Console

The repository of INTER-FW is in <https://git.INTER-IoT.eu/Inter-IoT/framework.git>. This repository contains the following folders:

1. **Global INTER-FW design.** It contains mock-up designs, source code engineering diagrams (reported in D4.3) and some additional resources useful for the application development. A detailed description of the main user interfaces in INTER-FW is also provided in the readme.md document.
2. **Front-end of the web application.** It provides the graphic user interface and overall navigation of the application
3. **Back-end of the web application.** It contains the back-end server, which supports the front end of the INTER-FW web application.

Figure 1 shows the list of commits in the INTER-FW source repository as of the date of submission of this deliverable, while **Figure 2** shows the list of files.

Autor	SHA1	Mensaje	Fecha
Miguel Llàcer	c9261ebbd4	universAAL entities support	hace 1 semana
Miguel Llàcer	e813cc569f	universAAL entities support	hace 1 semana
Miguel Llàcer	0730b1c9d6	app internationalization	hace 1 semana
Miguel Llàcer	d5cb4a9f4a	app internationalization	hace 1 semana
Miguel Llàcer	bb55fd8a56	login, status and services changes	hace 1 semana
Miguel Llàcer	a376a3f576	login, status and services changes	hace 1 semana
Miguel Llàcer	4d4122a1df	list and delete entities	hace 1 semana

Figure 1. INTER-FW GIT repository, list of commits

framework / design

- Miguel Angel Llorente Carmona (fbda0ae8c8) Users diagram added (hace 1 mes)
- resources (26eee12b0f) Updated INTER-FW design page (hace 2 meses)
- v0.2 (d7ed4a8183) new front-end mockup (hace 2 meses)
- v0.3 (d7ed4a8183) new front-end mockup (hace 2 meses)
- v0.4 (d7ed4a8183) new front-end mockup (hace 2 meses)
- v0.5 (85a7890678) New version of mockup and updated diagrams (hace 1 mes)
- README.md (fbda0ae8c8) Users diagram added (hace 1 mes)

INTER Framework design notes

Desing decisions, constants and other ideas around the webapp

Live online demo

Available in http://vmplsp04.westeurope.cloudapp.azure.com/interiot_wfk/ (Credentials are not required)

Known issues and backlog

- Login page: contact with a Id server. See D4.3 and D3.2 for further details.

Figure 2. INTER-FW GIT repository, directory structure

2.2 INTER-FW API Manager

The API Manager repository contains all the instructions, scripts and settings to:

- install the API manager for INTER-FW;
- Import/export settings;
- Import and publish APIs;
- Use the APIs.

In addition to the API Manager repository, an essential part for API definitions is the source repository that defines INTER-LAYER REST APIs, which is one of the outputs of WP3.

The list of INTER-API repositories is provided in Table 1.

Table 1. List of INTER-API related GIT repositories

Public git repositories	
API Manager	https://git.INTER-iot.eu/INTER-iot/api_manager
INTER-LAYER REST API definitions	https://git.INTER-iot.eu/INTER-iot/layer_apis

Figure 3 shows the list of commits in the INTER-API source repository as of the date of submission of this deliverable, while **Figure 4** shows the list of files.

The screenshot shows the GitHub repository page for 'Inter-iot / api_manager'. The repository has 44 stars and 0 forks. The commit history is displayed as follows:

Author	SHA1	Message	Date
Tomaz	2408a77355	Updated readme	1 week ago
Tomaz	40efb3bc08	Updated readme - bash code	1 week ago
Tomaz	782be1ea47	Updated readme	1 week ago
Tomaz	5b7ab0ed9b	Updated readme	1 week ago
Tomaz	57a86dae1a	API import/export tool	1 week ago
Tomaz	c952303119	Fixed bug, if not calling script from APIs directory.	1 week ago
Tomaz	43cc6341a4	Readme (preview)	1 week ago
Tomaz	99058b0868	APIs import script	1 week ago
Tomaz	e482b3d895	Exported APIs	1 week ago
Flavio Fuart	9120231b7e	basic info	1 week ago
Flavio Fuart	67584b49c4	readme.md	1 week ago

Figure 3. INTER-API GIT repository, list of commits

Inter-IoT / api_manager

Unwatch 44 Star 0 Fork 0

Files Issues 0 Pull Requests 0 Wiki

No Description

14 Commits 1 Branches 0 Releases

Branch: master api_manager

New file Upload file HTTPS SSH https://git.inter-iot.eu/Inter-IoT

File	Commit Hash	Commit Message	Time Ago
settings	57a86dae1a	API import/export tool	1 week ago
license.txt	ff039cffc5	Initial commit, readme.md in progress.	2 weeks ago
readme.md	2408a77355	Updated readme	1 week ago

readme.md

InterIoT API Manager deployment

This repository contains the instructions to deploy and setup the WSO2 API Manager for InterIoT.

Figure 4. INTER-API GIT repository, directory structure

3 Deployments of demo applications

3.1 INTER-FW Web Console

INTER-FW Web console is a web-based application, which unifies monitoring, configuration and management of different IoT components (sensors, network elements, gateways, platforms...) in a single environment at each interoperability layer. The Web console also includes key cross-layer management functions, such as configuration of authentication and/or authorization. It has eight different tabs providing different features:

- Platforms: instantiation, configuration and management of IoT platforms.
- Gateways: provision of functionalities at the gateway level.
- Network: management of virtual networks, QoS and networking rules.
- Semantics: definition of alignments, consultations with the semantic repository and basic operations upon the supported ontologies.
- API Management: linking to the INTER-API API Manager, managing level of access to the API, managing of the lifecycle, version and monitor statistics, among others.
- User management: a tool to manage the INTER-IoT users.
- Configuration: configuration of the INTER-FW tool.

The INTER-FW Web Console is publicly accessible at the following address:

http://vmplsp04.westeurope.cloudapp.azure.com/INTER-IoT_wfk/#

As the application implements dynamic rendering of tabs, several users are provided for testing purposes. The list of user names and their roles, as defined at the time of the submission of this deliverable, is provided in Table 2. Access (passwords) for testing purposes can be obtained on request.

Table 2. Web Console user names and roles

Details	User
See all the tabs	admin
Can't delete or modify	view_only
Semantic Engineer user	only_semantics
Network options are hidden	no_network
API management is hidden	no_apimanagement

Some screenshots of the applications are as follows:

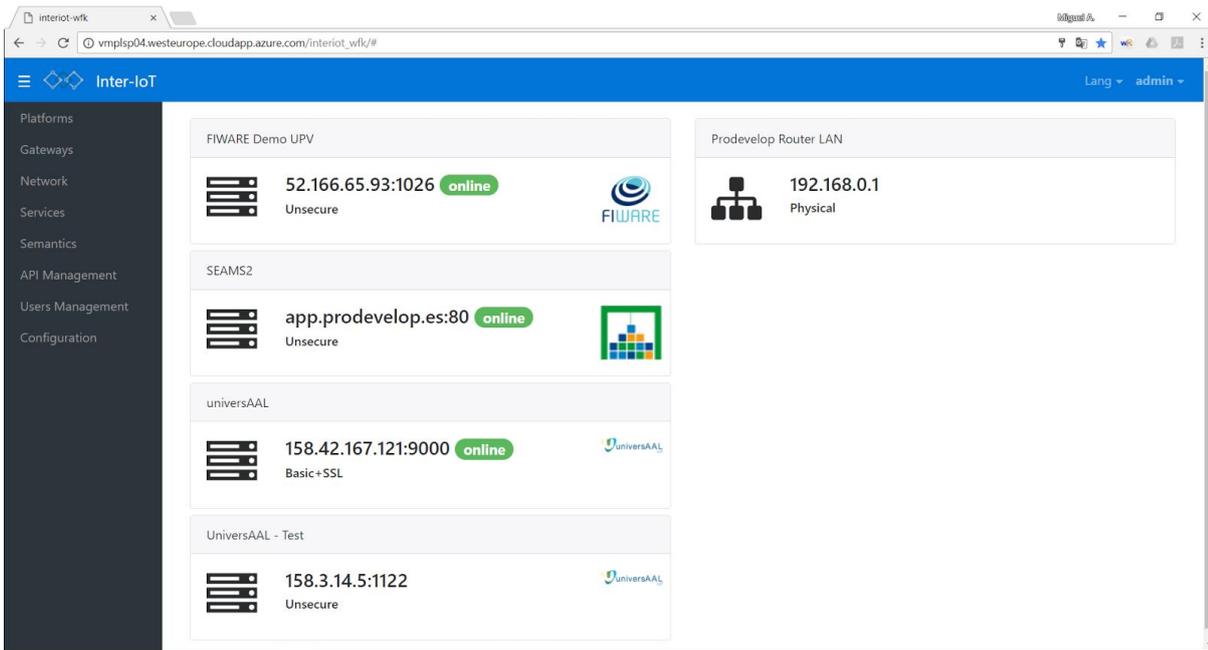


Figure 5. INTER-FW Web Console

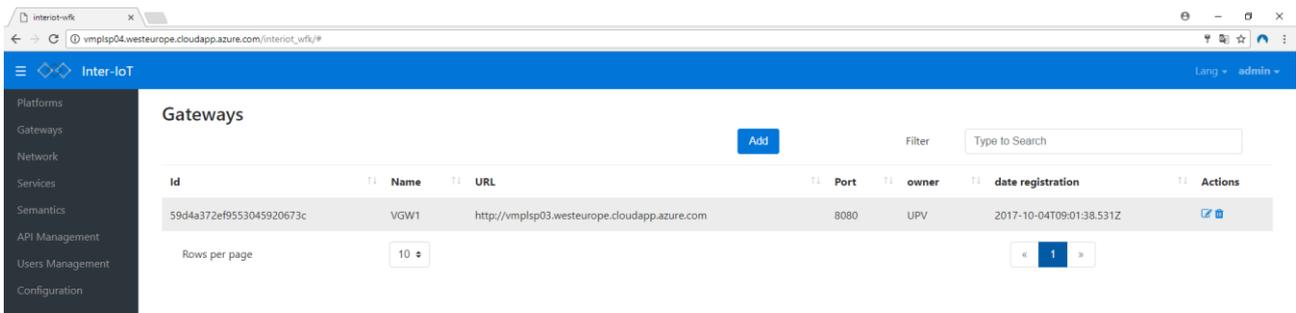


Figure 6 INTER-FW Web Console for Management and Listing of Gateways

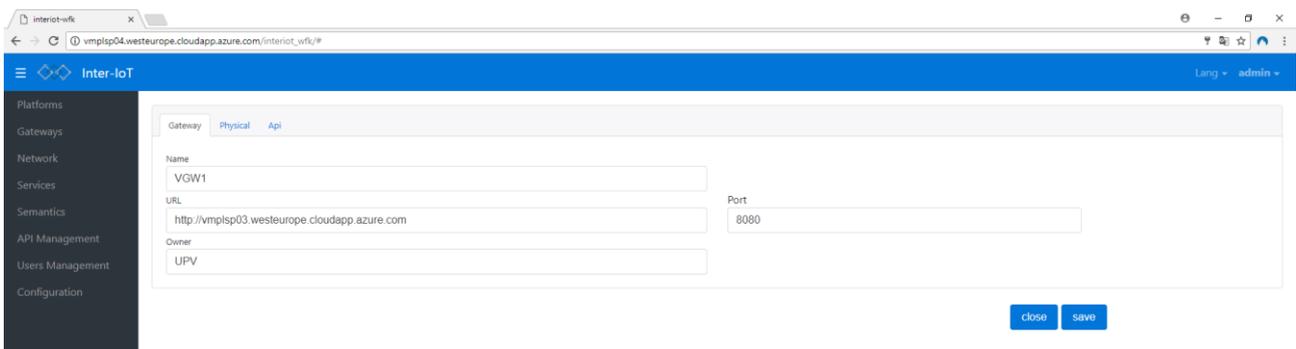


Figure 7 INTER-FW Web Console for access information on a Gateway

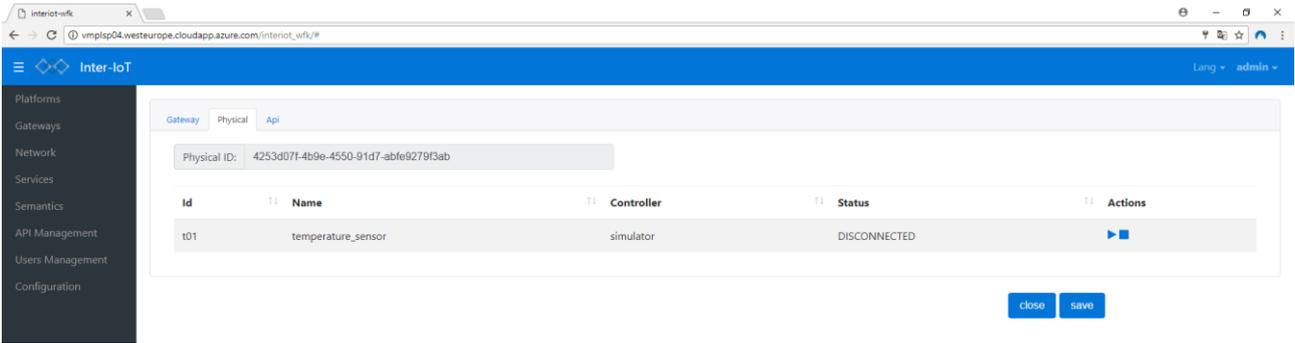


Figure 8 INTER-FW Web Console for management and listing of physical devices attached to the Gateway

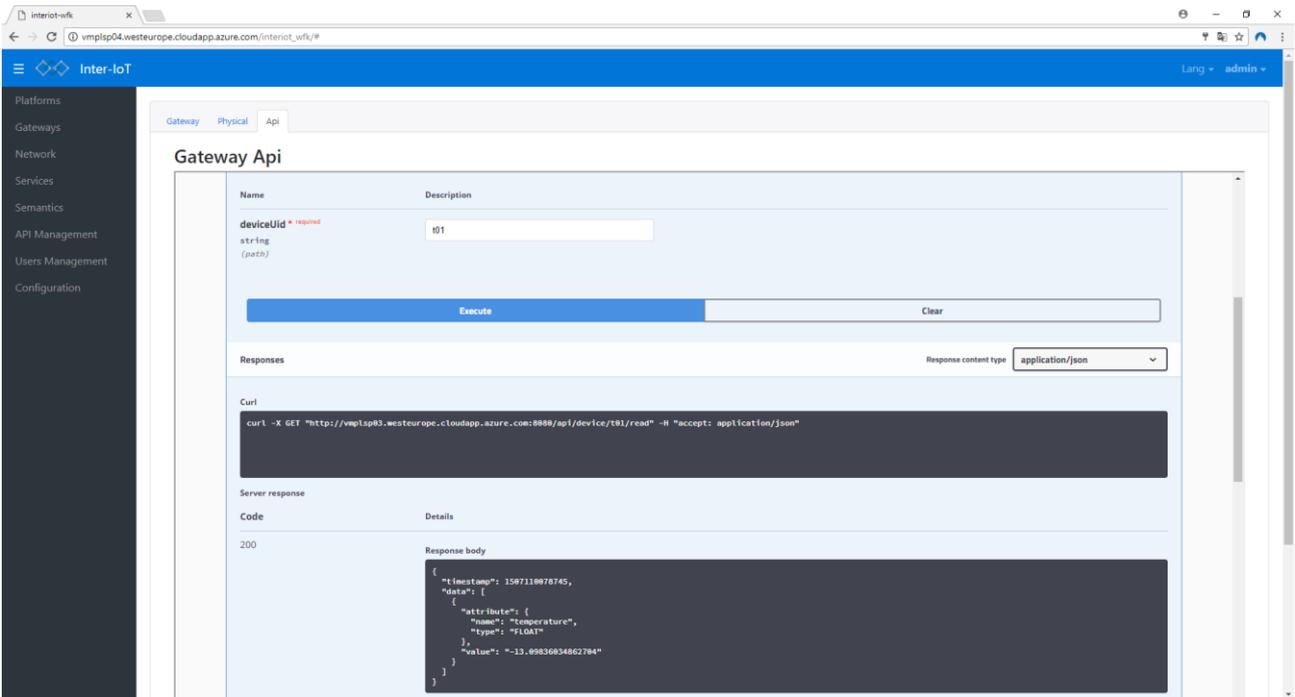


Figure 9 INTER-FW Web Console for accessing the HTTP interface of a Gateway

3.2 INTER-FW API Manager

We have deployed the WSO2 API Manager in the form of a Docker image on the Azure cloud virtual machine. This we did in order to facilitate the cloud deployment of INTER-FW.

The installation included the compulsory product registration (<https://wso2.com/user/register>), Docker image deployment and authentication with the WSO2 account. Appropriate URLs and API gateway endpoints have been configured as well.

The provision of unified API access through the API manager consists of several steps. First we created a API design document for each INTER-LAYER component, which is, in our case, provided through Swagger definitions. We then imported these definitions through the WSO2 API publisher interface.

The user should then subscribe to the APIs through the API subscription web GUI. In a specific INTER-IoT cloud deployment, we have identified the following types of API users with the corresponding set of access and management privileges:

- INTER-FW core users; users will full access to all INTER-FW features.
- INTER-FW front-end users; users with restricted set of access rights necessary to execute API calls exposed through the front-end interface.
- External users; at this point, there is no unified view of their level of access. This part will be defined in the following project phase with details to be provided in D4.4 and D4.6.

The INTER-API cloud version has been deployed to an Azure cloud instance with access links as defined in

Table 3.

Table 3. API Manager access links

Description	Access	URL
API Store	Public	https://vmplsp02.westeurope.cloudapp.azure.com/store
API Publisher	Restricted	https://vmplsp02.westeurope.cloudapp.azure.com/publisher
Management Console (Carbon)	Restricted	https://vmplsp02.westeurope.cloudapp.azure.com/carbon

The figures below provide screenshots of the API Manager Store and Publisher.

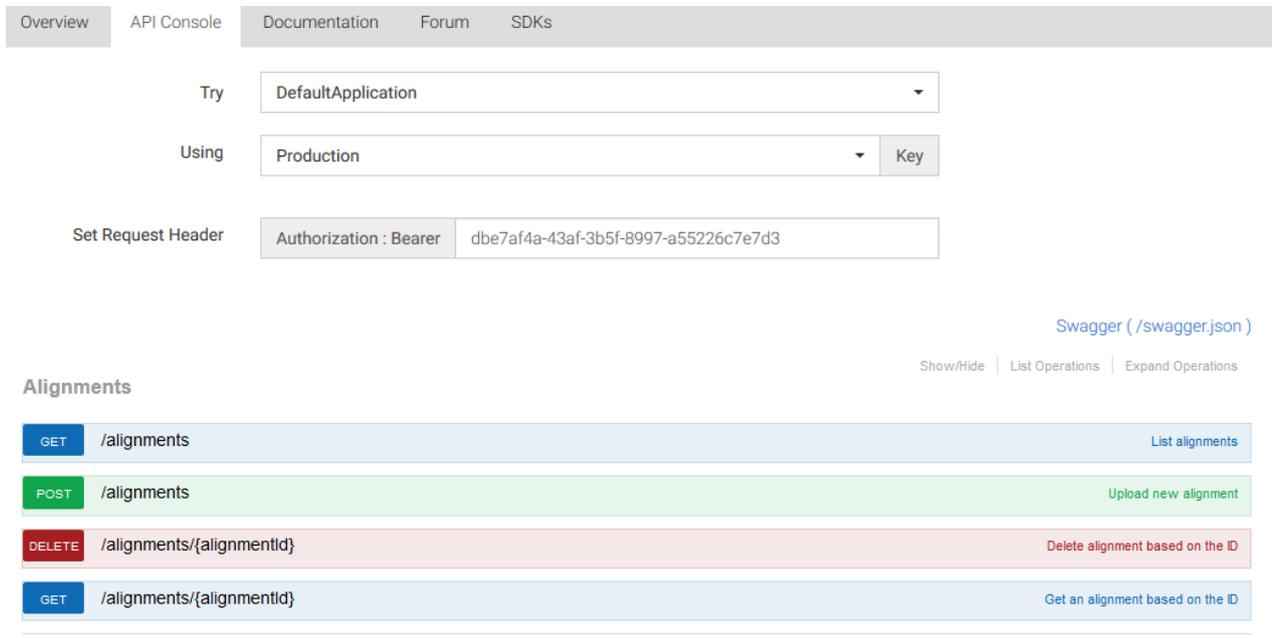


Figure 10. API Manager Store: API Console for testing APIs

API Name	Subscription Tier	Status	Actions
Inter-IoT_application_layer_interoperability - 1.0.0 PUBLISHED	Unlimited	UNBLOCKED	Unsubscribe
Inter-IoT_gateway - 1.0.0 PUBLISHED	Unlimited	UNBLOCKED	Unsubscribe
INTER-IoT_intermw - 0.0.1 PUBLISHED	Unlimited	UNBLOCKED	Unsubscribe
Inter-IoT_ipsm - 1.0.0 PUBLISHED	Silver	UNBLOCKED	Unsubscribe
Inter-IoT_network - 1.0.0 PUBLISHED	Unlimited	UNBLOCKED	Unsubscribe
Inter-IoT_semantic_repository - 1.0.0 PUBLISHED	Unlimited	UNBLOCKED	Unsubscribe

Figure 11. API Manager Store: Active subscriptions for a selected user

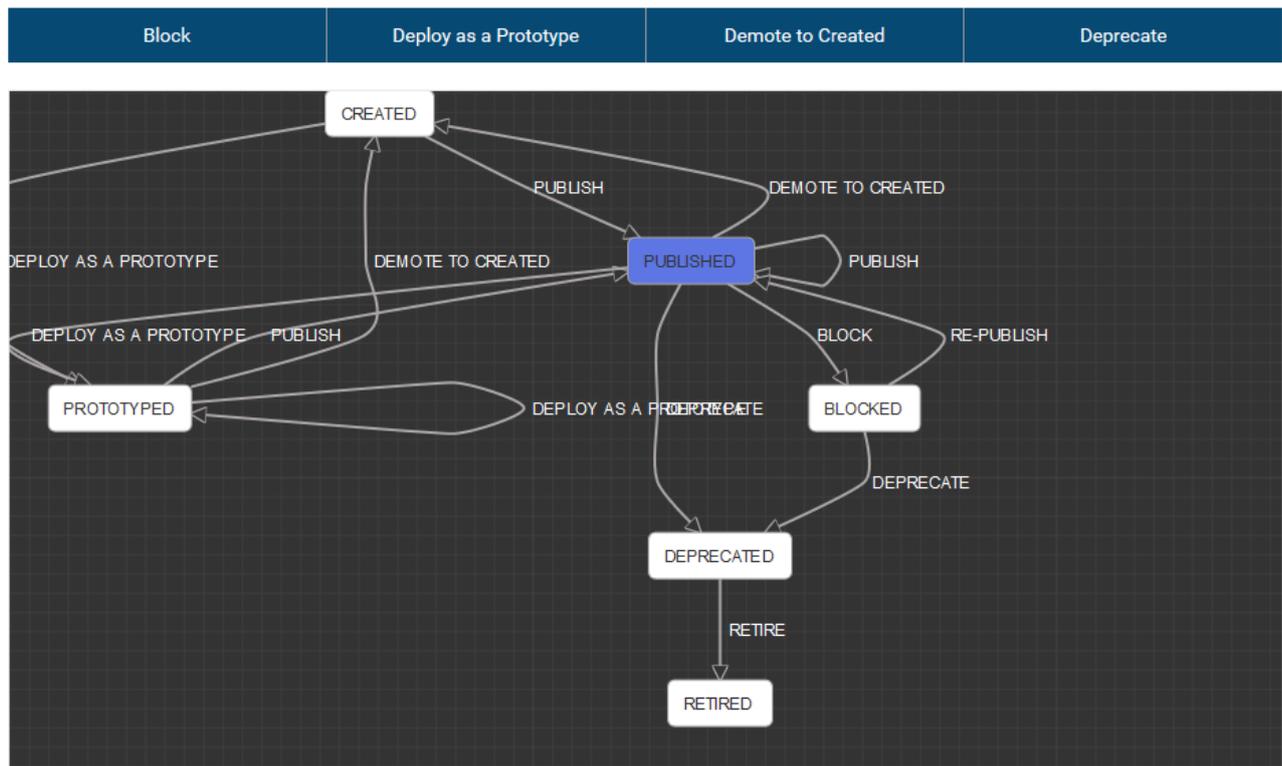


Figure 12. API Manager Publisher: An example of the API lifecycle management