On-boarding as a data provider

Version 1.1

Version History

Version	Date	Comment
1.0	01.01.2021	Initial Version
1.1	20.02.2024	Updated Versions and reviewed content

Introduction

iDDEN data exchange hub service is built using the ICAR Animal Data Exchange (ICAR ADE) standards for defining the API data contracts scheme and for the URL specifications as well.

The ICAR ADE standard is designed and defined by the ICAR technical working group at the public GitHub project.

The current release of the iDDEN hub service is implemented using the latest v1.3 version of the ICAR ADE standard and only for a few event types.

This document contains information on the technical aspects of integrating to the iDDEN system as a data provider.

Principles

ICAR ADE standard

In order to be able to communicate with the iDDEN Hub data exchange service, the data provider system developers and technical designers need to have a good understanding of ICAR ADE standard principles for both message definitions and URL specifications. The iDDEN functionality on the API layer and on the data integration layer is built using the ICAR ADE standard.

ICAR ADE standard is not only used on the iDDEN API layer but also on the integration side of messages routing to the data provider server interfaces. Request content is routed through the iDDEN Hub in the original structure and is sent to the receiver in ADE format.

Refer to the official ICAR ADE project at GitHub in order to get full details of the standard and a roadmap for the upcoming work.

Most developers should be familiar with REST and JSON files. To simply view the standard, you can open the example schema file in your favourite OpenAPI Specification editor. For example, open it in the popular Swagger Editor.

General data exchange flow

Step	Action
1	Client application invokes the right web service method for executing the action it needs to make.
2	Client application performs an authentication logon to the data provider authentication service. The returned issued security token will be used to authorize communication with iDDEN and the data provider systems.
3	iDDEN hub service checks that the client has the right to use the web service and if so, checks data integrity: • If the message is well-formed and authentication is okay, proceed. • Otherwise, an error message with an explanation in the user's language is sent as a response. The response ends this use case.
4	iDDEN checks the data provider that is used as a data exchange target and selects the respective adapter to process the request to the data provider interface. Additional adapters might be used if conversion is needed for different versions of the ICAR ADE standard messages.
5	Data provider service receives the request from iDDEN. The data provider system

	will authenticate the request and then process it: • If authentication or authorization fails, an error explanation in the user's language is sent as a response • If the data provided by the client is incorrect or incomplete in any way, an error explanation in the user's language is sent as a response • If the data provided by the client is correct and authentication / authorization exists, a response to the request is formed in the data provider system
6	iDDEN receives the response from the data provider system.
7	iDDEN sends the response to the client
8	Client application processes the response according to its own inner logic.

Integration with iDDEN

In order to be integrated with the iDDEN solution, each registered data provider needs to develop and provide its own ICAR ADE compatible data service for data exchange.

The information needed for data provider integration & acceptance testing environments is listed below.

- The physical addresses of the data provider service APIs in each environment (they need to be visible to the internet).
- Test user credentials needed to call the data provider service API in each environment plus instructions on how to obtain a security token using those credentials. As we also mentioned in the technical notes, the credentials themselves will not be passed through iDDEN but the authentication will be a separate action outside of the scope, producing a security token, and THAT token is passed within the iDDEN calls. Please note that in the integration test environment, you can also use one or more fake tokens, without need for actual authentication.
- Contact information to those technical persons in the data provider projects that our developers & testers can directly ask for clarifications during the development, as well as agree upon the test cases used in each environment. The most important thing here is that there is in each environment a person available who knows in detail how the data provider service API works, so that we can avoid unnecessary delays and "message forwarding" between persons in the iDDEN project and data provider projects.

For more information please refer to the "iDDEN – Organization registration" document.

ICAR ADE data provider service template

iDDEN is providing a technical project template for easy base setup for the ICAR ADE compliant data provider API service. The project template produces ready-to-run projects that make it easy for users to start with a functional set of code. The template is delivered as a template pack via the NuGet package.

Technology stack:

- **.NET 6** for the Web API project
- **.NET Standard 2.1** for the ICAR ADE library project
  - Model generated from the latest v1.3
- **Swashbuckle.AspNetCore**-library included for Swagger documentation for the API
- **Newtonsoft.Json** -library is used for JSON data procession

Usage:

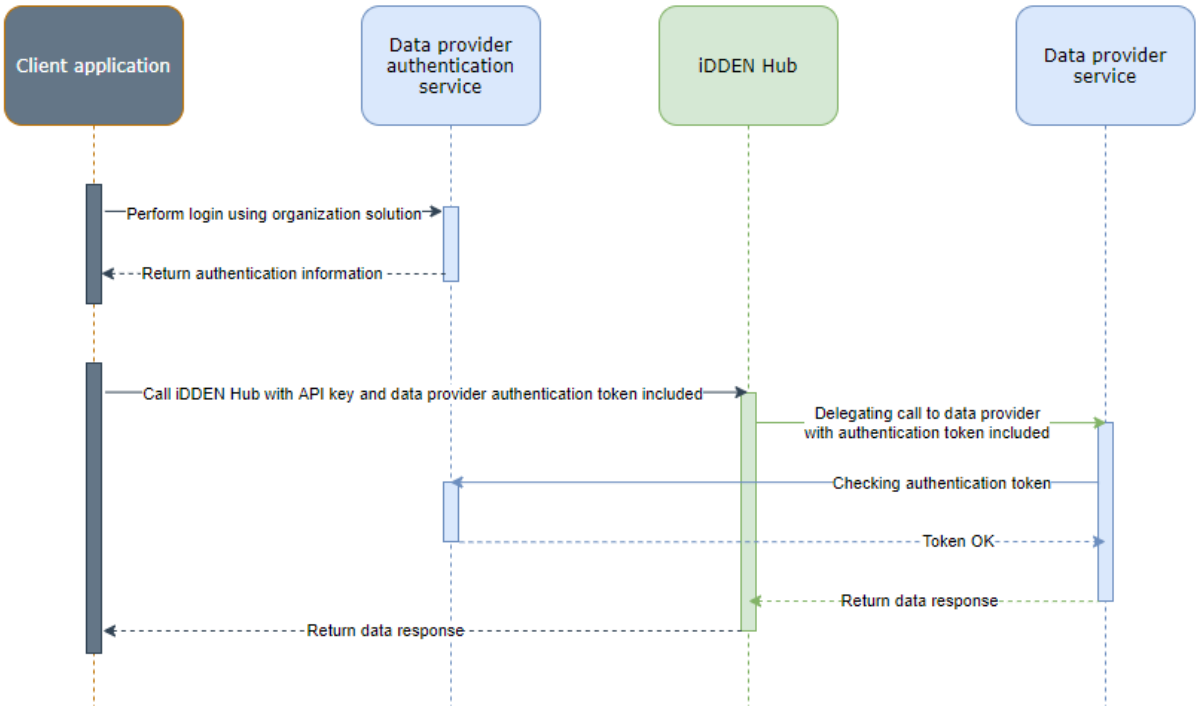
- Run the command for installing the template “dotnet new -i {path-to-template-pack}
  - *dotnet new -i DataProvider.Templates.1.0.1.nupkg*
- Run the command for creating a project using the template “dotnet new {project-template-name}
  - *dotnet new adedataprovider*

Authentication and authorization on data provider

iDDEN –service security solution follows these general principles:

- IDDEN does not perform any end user authentication/authorization logic itself, but it assumes the authentication will be implemented by each organization backend system.
- iDDEN assumes the organization authentication produces a single string-type authentication token which is included within the HTTP header “Authorization” in the iDDEN requests. The format of this token technically can be anything as the iDDEN only passes it through and does not parse or process it in any way (ASCII-7bit characters are assumed for compatibility reasons).

This basic principle is illustrated in the picture below.



(Picture 1. iDDEN authentication principle)

For more information, please refer to the “iDDEN – Security handling” document.

iDDEN available ADE Messages

Please refer to the “iDDEN – Messages” document.

Related documentation

1. iDDEN – ID
2. iDDEN – Organization registration
3. iDDEN – Client authentication and data routing
4. iDDEN – System description

5. iDDEN – Use cases
6. iDDEN – Security handling
7. iDDEN – ICAR ADE information
8. iDDEN – Messages