iDDEN

The international dairy data exchange network

System Description

Version 1.0

Version History

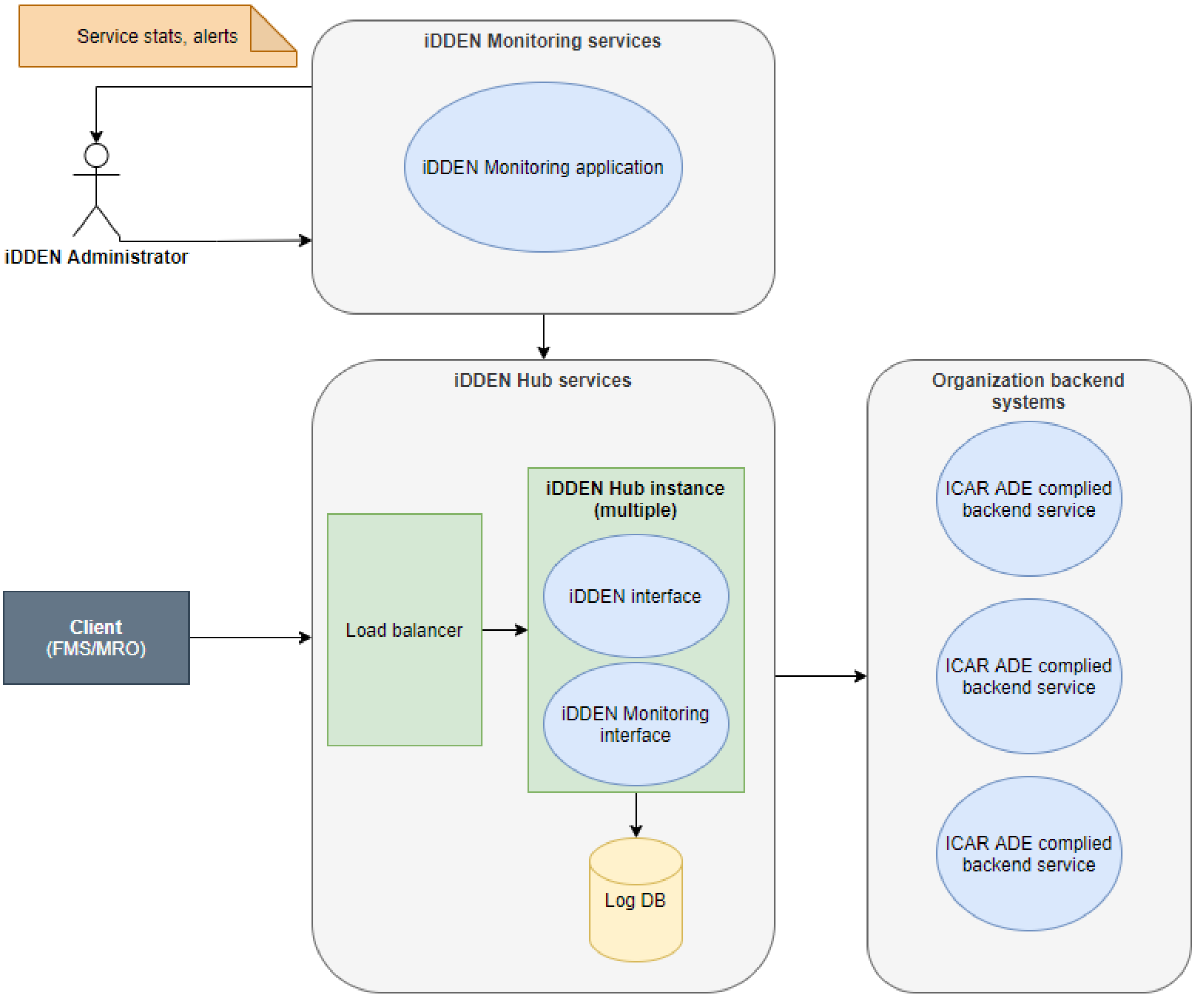
Version	Date	Comment
1.0	01.01.2021	Initial Verison

System components

The iDDEN Hub is an open interface solution to exchange cattle data between management systems. iDDEN consists of the following components:

- **Client applications**
  - Farm management systems (on-site/on-farm) exchanging cattle data through the iDDEN Hub service with the services of a data recording organisation/ processing centre.
  - Data recording organisation/processing centre client services exchanging cattle data through the iDDEN Hub service with farm management cloud systems.
  - (potentially) Farm management cloud client services exchanging cattle data through the iDDEN Hub service with the services of a data recording organisation/ processing centre.
- **Routing service**
  - Main component of the iDDEN solution is the public HUB interface. Technically, it is a REST -interface that uses JSON as data format and is built on ICAR standardised data contract messages (ICAR ADE).
  - The service is developed using ASP.NET Core framework.
  - The web service has several roles:
    - It provides the ICAR ADE standardised data contracts for client applications.
    - It hides all communication details and address information of the cattle database systems from the clients.
    - It validates incoming request to accept only iDDEN authorized client connections. Each request should include a valid client API key to be routed.
    - It routes the service calls from the client applications to a data provider web service hosted by the target organisations. Client applications must always point out the target organisation for the request in the header part of the request data object.
- **Data provider web services**
  - The service requests sent through the iDDEN Hub web service are then routed to data provider web services hosted by different organisations that are responsible for the communication with the local cattle databases. The data provider services are responsible for:
    - Authorising the original request (validating the authentication/token information passed within the original HTTP request) from the client.
    - Checks farm authorization mandates for data exchange between recording organizations and cloud-based farm management systems.
    - Validating the request and making the necessary data operations against the local databases.
- **Log database**
  - iDDEN Hub web service logs all the requests and the responses from the data provider web services for troubleshooting scenarios for limited time (could be defined per domain).
- **Administration tool**
  - Administration/monitoring UI provides tools to monitor iDDEN statuses and specially to analyse the content of the iDDEN log database.

A basic communication diagram of the iDDEN Hub web service and the national web services is shown below. There are two instances of the main service, located at different servers. The secondary instance acts as a backup instance, if the primary instance is not available for some reason.


Service stats, alerts
iDDEN Monitoring services
iDDEN Monitoring application
iDDEN Administrator
iDDEN Hub services
iDDEN Hub instance (multiple)
Load balancer
iDDEN interface
iDDEN Monitoring interface
Log DB
Client (FMS/MRO)
Organization backend systems
ICAR ADE complied backend service
ICAR ADE complied backend service
ICAR ADE complied backend service

Authentication and authorisation

iDDEN requires two kinds of authentication information:

- Client authentication information
  - Before a client system can start to use the iDDEN Hub interface, it must possess an API key and official identifier (iDDEN ID) that verifies the calling party.
  - The API key is a stable string value that is generated by the iDDEN authority for each client system registered to the iDDEN system that intends to use the Hub service.
  - The API key is much like a client certificate.
- End user authentication information
  - The iDDEN service itself does not have any knowledge of the details of the data provider system user authentication/authorisation solution, since iDDEN Hub acts only as a message router between the client and data provider systems. However, there is an agreement that each national end user authentication solution must:
    - Accept the end user authentication token in the HTTP header within iDDEN service requests.

- The iDDEN Hub passes the token further to the organisation backend service (unmodified) and the national service takes the responsibility to validate the token and makes the other necessary operations required to authorise the end user.

IDDEN ID

IDDEN ID is a unique piece of identification information provided to each organization registered to the iDDEN system. The format of the iDDEN ID is described in full detail in the additional documentation (iDDEN ID). The iDDEN-ID will be sent out to any partner during the registration process (see the iDDEN Organization registration document).

iDDEN ID should be provided in the http request inside a non-standard http header using the following format:

iDDEN-ID: [identifier delivered by iDDEN support]

IDDEN API Key

The iDDEN support authority generates the API Key for each individual client system and delivers it to the contact person(s) responsible for the support, marketing and/or development of that system.

The API key must be delivered within each iDDEN service request, placing it inside a custom HTTP header named "**iDDEN-API-Key**" using the following format:

iDDEN-API-Key: [API key delivered by iDDEN support]

The API key is strongly related to the "iDDEN-ID" data delivered in a separate non-standard header of the iDDEN Hub requests (see sample below), so the iDDEN identifier of the client system should never be changed after an API key has been generated for it.

End user authentication

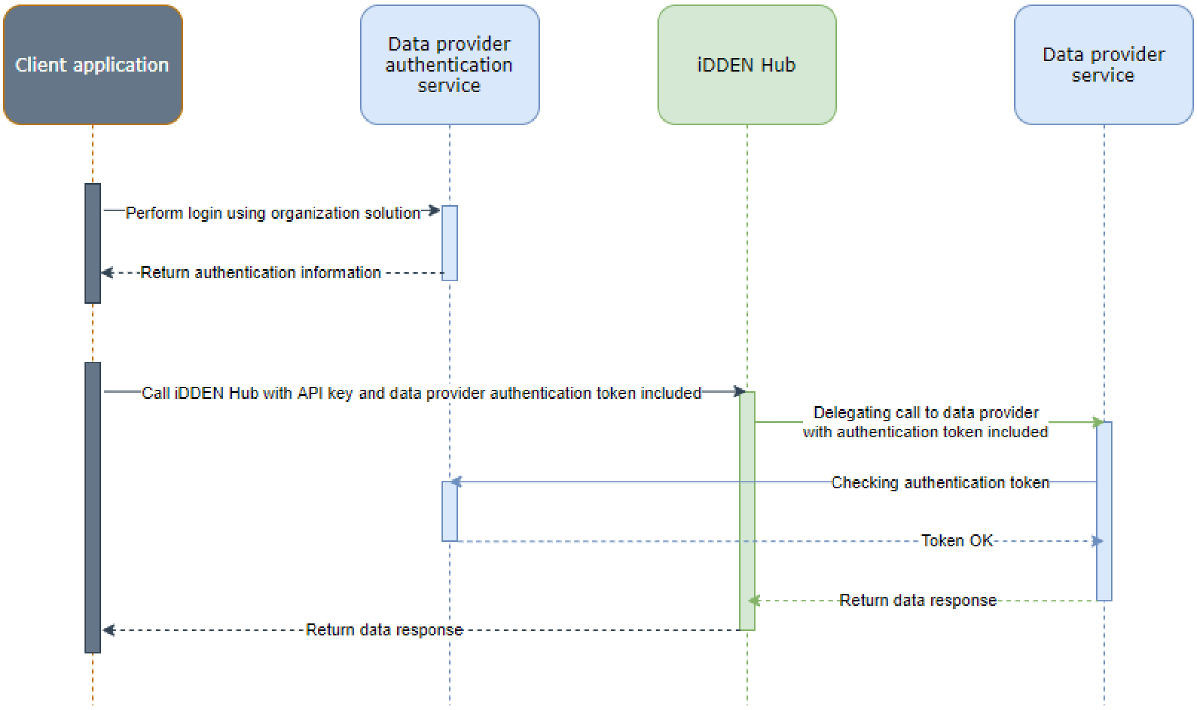
End user authentication solutions used for obtaining the end user authentication token are completely specific for each target system and the organisations are also responsible for providing the support for client services concerning their own solution.

After the successful reception of the authentication token from the target authentication system of the data provider, it is always delivered within the iDDEN service requests in a standard place, placing it inside "**Authorisation**" HTTP header, using the following format:

Authorization: [authentication token value, received from national authentication solution]

When authenticating against the Finnish service, an authentication scheme named 'TOKEN' must precede the token value in header.

You can see below a simplified, high-level diagram that wraps together the whole authentication process flow in case of a successful authentication.


Client application
Data provider authentication service
iDDEN Hub
Data provider service
Perform login using organization solution
Return authentication information
Call iDDEN Hub with API key and data provider authentication token included
Delegating call to data provider with authentication token included
Checking authentication token
Token OK
Return data response
Return data response

There is a “real”-looking raw iDDEN Hub HTTP request further below.

- **iDDEN-ID (http header)**
    - iDDEN identifier for the client system
- **iDDEN-API-KEY (http header)**
    - iDDEN API key generated for the client system defined in the iDDEN-ID header
- **iDDEN-ID-TARGET (http header)**
    - iDDEN identifier of the target system which the client wants to use as data provider
- **Authorisation (http header)**
    - Original token-based information received from the data provider authentication service
- **HTTP request content**
    - request content which need to be routed to the receiving data provider system

```
GET https://hub.idden.net/v1/fin.herd/303063 HTTP/1.1
Accept-Encoding: gzip,deflate
Content-Type: application/json
iDDEN-ID: DEU-MRO-518853215841
iDDEN-ID-TARGET: INT-OCP-548544158458
iDDEN-API-KEY: bpQWgYc2fZh6LyJyfDn1HAfELMM=
Authorization: Bearer
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwia
WF0IjoxNTE2MjM5MDIyfQ.SflKxwRJSMeKKF2QT4fwpMeJf36POk6yJV_adQssw5c
Content-Length: 252
Host: hub.idden.net
Connection: Keep-Alive
User-Agent: Apache-HttpClient/4.1.1 (java 1.5)

{
   "start-date-time": "2014-01-13",
   "end-date-time ": "2016-01-13"
}
```