Sadevio On-Premise configuration

This section will focus on how to configure the SADEVIO system to activate the Velocity integration when running on-premise.

Configure the access system

- **Create a New Access System**:
  - Go to **Maintenance**.
  - Select **Access System**.
  - Click on **Create New Access System**.
- **Select Connector Type**:
  - Choose **Identiv Velocity** as the connector type.
- **Select Time Conversion**:
  - Choose use local time.

image.png
Image not found or type unknown

Configure the Velocity REST API

- **Set REST API URL**:
  - Enter the IP address and port for the Velocity REST API. e.g. http://192.168.1.61:8088
- **Enter Credentials**:
  - Provide the password generated with the Velocity configuration tool. See Generate API Password
- **Configure Card Format**:
  - Select the card format algorithm you want to use to generate the Velocity MATCH code. Refer to the Velocity Card Format Mapping section for details.
- **Set Credential Template ID**:
  - Enter the ID of your generated credential template in Velocity. Refer to the Configure Velocity Badge Template section for more information.

Configure User Defined Fields

- **Map SADEVIO Fields to Velocity Fields**:
  - Under the **User Defined Fields** section in the configuration, you can map SADEVIO fields to the corresponding user-defined fields in Velocity.
- **Populate Velocity Fields**:
  - The SADEVIO system will automatically populate the mapped information into the appropriate Velocity fields.

image.png
Image not found or type unknown

Configure Custom Card Data Transformation

In the **UID Post Processor** section, you can add custom JavaScript code to manipulate the actual transferred credential value. This is useful if you need to change the order of a card value or add a facility code in a QR code scenario.

The original value is stored in the variable `cardNumber`. Your code should return a value that represents the new credential value.

Example: Adding a Facility Code

The following example checks if the provided UID is 6 characters long. If it is, the code takes the last 5 characters and appends a prefix of `1755-` to the value.

```
if(cardNumber != "" && cardNumber.length == 6 && cardNumber.startsWith('0')){
   var newUid = '1755-'+cardNumber.substr(1);
   return newUid;
}else{
   return cardNumber;
}
```

Place this code in the **UID Post Processor** section to achieve the desired transformation.

Revision #5
Created 28 June 2024 00:10:17 by Admin
Updated 28 June 2024 00:52:39 by Admin