Use Cases and Receipts

In this chapter, we will explore the best practices for working with the Tydids framework. Through practical examples and real-life scenarios, you will learn how to effectively use the framework’s validation and cryptographic processes. This chapter aims to equip you with the knowledge to seamlessly integrate Tydids into your projects, ensuring robust security and reliable performance.

- Lending a drilling machine
- Getting freelancer work signed off and paid
- How ACME Corp uses Self-Sovereign Identity (SSI) to give customers full control over their personal data
- Check Grant Status via REST API

Lending a drilling machine



In this scenario, Alice, a homeowner, wants to lend her drilling machine to Bob, a neighbor who needs to perform some home improvement tasks. To ensure transparency and trust, Alice will receive a digital receipt confirming that she has lend the drilling machine from Alice.

What should be achieved?

Transparency and trust between Alice and Bob about the lending of a drilling machine.

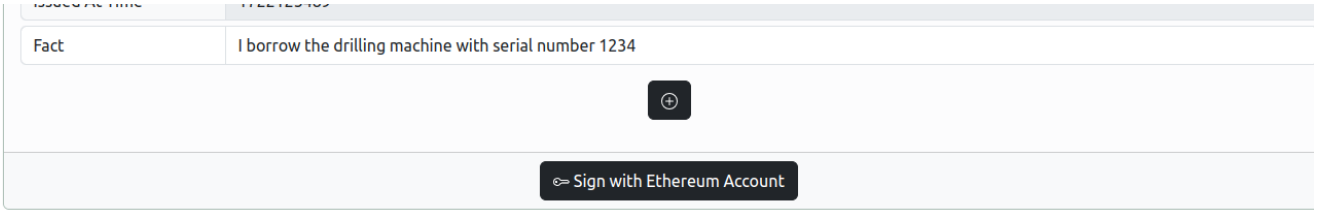
What should not happen?

- Alice and Bob do not want to make their real identities transparent to others.
- The fact that it is a lending process should not be exposed.
- The fact that a drilling machine should not be exposed.

How to do this?

For simplicity, both Alice and Bob know the ID of each other from the past. *(Hint: You get your ID by clicking on the upper right Icon. It will give your ID and allow you scanning the ID of someone else to see it).*

- Bob opens https://tydids.com/
- Creates a new validation
- Using the "+" adds a field named "Fact" with the value "I borrow the drilling machine with serial number 1234"



- Bob signs the validation and gets a PDF document that he will give to Alice.
- Alice opens the Verifier of Tydids and validates the signatures and gives the drilling machine to Bob.

Important: Do only trust if you trust

Basic rule: "Zero Trust"

The example has a single point where trust is needed: It relies on the past/history knowledge of the IDs (=*consensus*). But what if this does not exist? What if Bob just moved to the new house? A technical answer to this is "strong qualified signatures".

Getting freelancer work signed off and paid



Alice works for a multinational corporation and needs to sign a new consulting agreement with Bob, a contractor providing services to her company. The agreement outlines the terms of Bob's freelance work, and payment should be made directly to his designated wallet.

eIDAS in a Nutshell

The eIDAS (electronic IDentification, Authentication and trust Services) regulation simplifies secure electronic interactions within the European Union. It ensures businesses, citizens, and governments can interact digitally with standardized identification methods and trust services – all seamlessly working across EU member states. This regulation promotes trust in electronic transactions, fostering a more integrated digital market within the EU.

Goal

To securely sign the consulting agreement in a way that can be verified by a third party if necessary.

How to do this?

1. **Bob Prepares the Contract:** Bob creates and shares the contract electronically, attaching it to a validation request.
2. **Alice Signs the Contract:**
   - Alice visits https://tydids.com/.
   - She creates a new validation request.
   - Uploads the contract as a PDF document (using the "Attach" button).
   - Signs the document electronically.

- To ensure highest security, she then uses the "Request eID validation" button to finalize the signing with her electronic ID (eID).

3. **Verification and Payment:**
   - Once signed, Alice shares the eID-validated document with Bob, who can view it under "Verifications" in the online system.
   - Upon confirmation of the agreement and completion of Bob's services, Alice's company uses the verified eID information to transfer the payment directly to Bob's designated wallet.

How ACME Corp uses Self-Sovereign Identity (SSI) to give customers full control over their personal data

Showcase: https://energychain.github.io/tydids-validation/public/DataIdentity/

Scenario

Experience how ACME Corp. uses Self-Sovereign Identity (SSI) to give you full control over your personal data. With SSI, you can manage and revoke consent anytime, ensuring compliance with GDPR and other privacy regulations.

Key Benefits

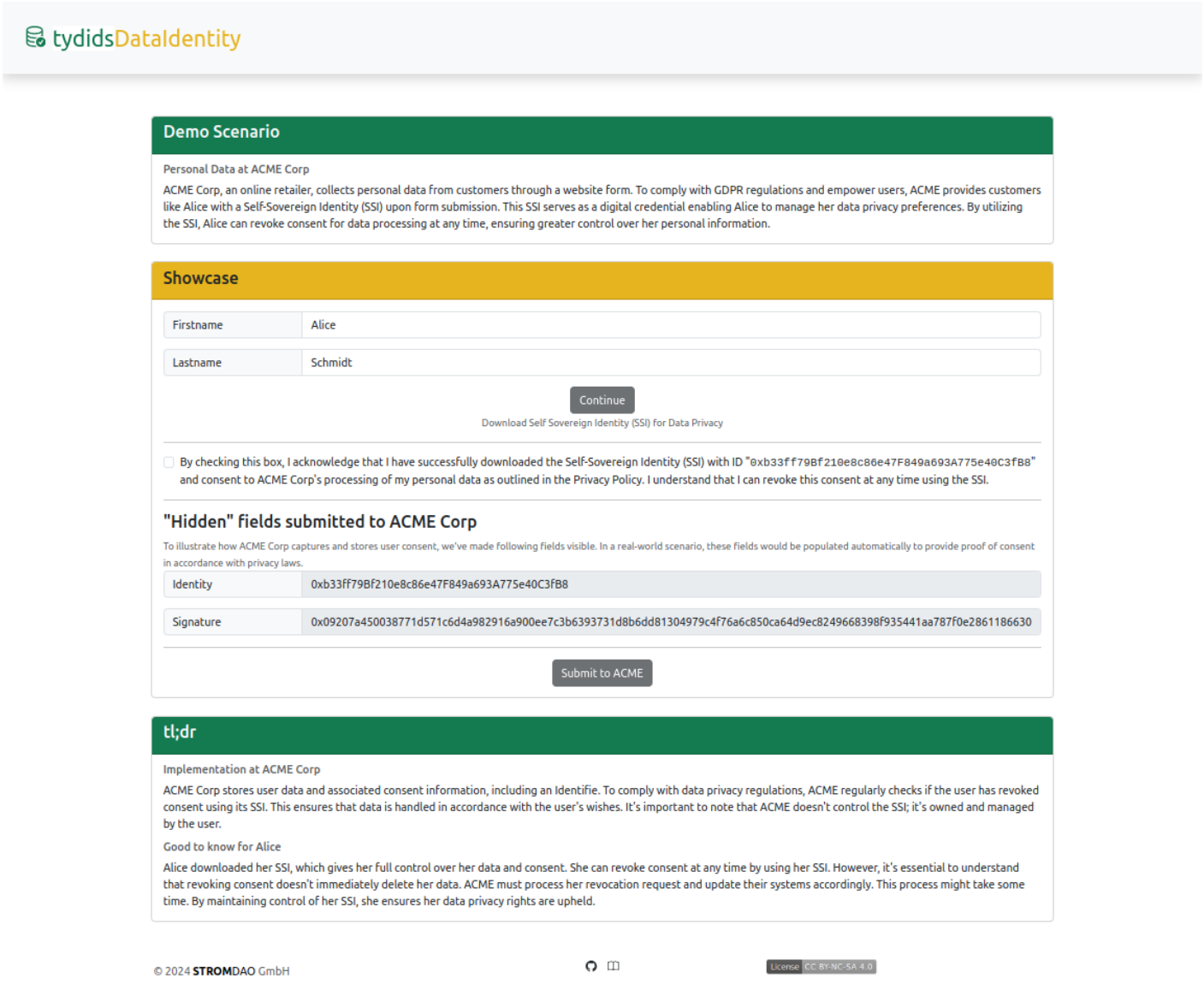
- **Full Data Control:** Download your SSI to manage consent preferences easily.
- **Privacy Compliance:** Ensure your data is handled according to GDPR and other privacy regulations.
- **User Empowerment:** Maintain ownership of your data and revoke consent at your discretion.
- **Seamless Integration:** ACME Corp. processes your revocation request efficiently, maintaining data privacy and security.

Demo Instructions

1. **Download Your SSI:** Click "Download Self Sovereign Identity (SSI) for Data Privacy."
2. **Acknowledge and Consent:** Check the privacy checkbox to consent to ACME Corp.’s processing of your personal data.
3. **Submit Your Data:** Click "Submit to ACME" to see how your consent is securely handled.

Why It Matters

By using SSI, you have unparalleled control over your personal information. Protect your privacy, ensure regulatory compliance, and experience the future of data consent management with TydidsDataIdentity.



Check Grant Status via REST API

This document describes a simple REST API call to check the revocation status of a specific identity on Currently.io. This functionality is useful for organizations that implement the TyDIDS Trust Framework for GDPR compliance.

API Endpoint

- **URL:** https://api.currently.io/v2.0/tydids/status?identity=<identity>
- **Method:** GET
- **Path Parameter:**
  - `<identity>`: Replace this with the identity you want to check the revocation status for. The identity is typically an SSI (Self-Sovereign Identity) identifier string.

Response

The API responds with a JSON object containing information about the revocation status and consensus value. The HTTP status code of the response indicates the overall status of the request.

- **HTTP Status Code 200 (OK):**
  - **Body:**
    JSON

    ```
    {
      "status": "granted",
      "consensus": 40856
    }
    ```

    Verwende den Code mit Vorsicht.
  - **Description:** The identity has **not** revoked data approval.
    - `status`: String indicating the approval status. In this case, the value is `"granted"`.
    - `consensus`: Integer representing the number of the latest consensus reached at the time of the request.
- **HTTP Status Code 403 (Forbidden):**

- **Body:**
  JSON

```
{
  "status": "revoked",
  "consensus": 40850,
  "revoked": 1722722789
}
```

  Verwende den Code mit Vorsicht.

- **Description:** The identity has **revoked** data approval.
  - `status`: String indicating the approval status. In this case, the value is `"revoked"`.
  - `consensus`: Integer representing the number of the latest consensus reached at the time of the request.
  - `revoked`: Integer representing the timestamp (in Unix time) at which the approval was revoked.

Example Usage

- **Checking an identity that has not revoked data approval:**
  - API Call:
    https://api.corrently.io/v2.0/tydids/status?identity=0x95Bee09c395c60883Fa8bb95F05404a71f7ee7F7
  - Response Body:
    JSON

```
{
  "status": "granted",
  "consensus": 40856
}
```

    Verwende den Code mit Vorsicht.

- **Checking an identity that has revoked data approval:**
  - API Call:
    https://api.corrently.io/v2.0/tydids/status?identity=0xa8CD7c57c144be63852Da3C44D97088A740D43Cd
  - Response Body:
    JSON

```
{
  "status": "revoked",
  "consensus": 40850,
  "revoked": 1722722789
}
```


Verwende den Code mit Vorsicht.

GDPR Compliance

Organizations implementing the TyDIDS Trust Framework for GDPR can use this API to regularly check the revocation status of identities. If an identity revokes data approval, the organization can take appropriate actions, such as stopping data processing or informing the user.