Standard Operating Procedure - TOKEN-BASED AUTHENTICATION

AU.2.E

**Version Control**

|  |  |  |
| --- | --- | --- |
| Version | Date | Changes Made |
|  |  |  |
|  |  |  |
|  |  |  |

**Guidelines for Maintaining the SOP Version Control Table:**

* **Version**: Assign a new version number for every update. Minor changes can be denoted by incremental changes in decimal (e.g., 1.1, 1.2), while major changes can increment the whole number (e.g., 1.0 to 2.0).
* **Date**: The date when the changes were finalised.
* **Changes Made**: A brief description of the changes or updates made.

Table of Contents

[1. Purpose 3](#_Toc177285353)

[2. Definitions and Abbreviations 3](#_Toc177285354)

[3. Application 3](#_Toc177285355)

[3.1 Ownership and Stakeholders 3](#_Toc177285356)

[3.1.1 Digital Identity Service Providers (DISPs) 3](#_Toc177285357)

[3.1.2 IT and Security Teams 3](#_Toc177285358)

[3.1.3 Compliance and Legal Departments 3](#_Toc177285359)

[3.2 Users and Beneficiaries 3](#_Toc177285360)

[3.2.1 General Public 3](#_Toc177285361)

[3.2.2 Government Agencies 4](#_Toc177285362)

[3.2.3 Private Sector Companies 4](#_Toc177285363)

[4. Prerequisites 4](#_Toc177285364)

[4.1 Assumptions 4](#_Toc177285365)

[4.2 Constraints 4](#_Toc177285366)

[5. Process Flow - Process and Procedures 4](#_Toc177285367)

[5.1. Initiating Token-Based Authentication: 4](#_Toc177285368)

[5.2. Sending Token and Credentials for Authentication: 4](#_Toc177285369)

[5.3. Verifying Credentials and Token: 5](#_Toc177285370)

[5.4. Handling Successful Authentication: 5](#_Toc177285371)

[5.5. Handling Unsuccessful Authentication Attempts: 5](#_Toc177285372)

[5.6. Logging and Status Update: 5](#_Toc177285373)

[5.7. Process Termination: 6](#_Toc177285374)

[6. Visualisation 7](#_Toc177285375)

# 1. Purpose

This SOP outlines the standardised procedure for token-based authentication within the Digital Identity (DID) system. It ensures secure and accurate authentication through proper verification, encryption, and error handling.

# 2. Definitions and Abbreviations

**DID**: Digital Identity

**KM**: Key Manager

**KR**: Key Revocation

**HSM**: Hardware Security Module

**CA**: Certificate Authority

**IDA**: ID Authentication Database

**AC**: Access Control

**FTP**: First Time Password

**HTTPS**: Hyper Text Transfer Protocol Secure

# 3. Application

## 3.1 Ownership and Stakeholders

### 3.1.1 Digital Identity Service Providers (DISPs)

* **Ownership**: Oversee the token-based authentication process.
* **Responsibilities**: Ensure secure and compliant authentication using tokens.

### 3.1.2 IT and Security Teams

* **Ownership**: Manage technical infrastructure and security protocols.
* **Responsibilities**: Maintain system security, data encryption, and infrastructure.

### 3.1.3 Compliance and Legal Departments

* **Ownership**: Ensure compliance with legal and regulatory standards.
* **Responsibilities**: Oversee compliance checks, documentation, and regulatory adherence.

## 3.2 Users and Beneficiaries

### 3.2.1 General Public

* **Users**: Individuals using tokens for authentication.
* **Usage**: Provide tokens and credentials for secure authentication.

### 3.2.2 Government Agencies

* **Users**: Agencies requiring verified identities for services.
* **Usage**: Utilise verified identity information for secure service delivery.

### 3.2.3 Private Sector Companies

* **Users**: Businesses requiring high-security identity verification.
* **Usage**: Use secured identities for compliance and verification purposes.

# 4. Prerequisites

## 4.1 Assumptions

* Subscribers and verifiers have received their DID and FTP credentials.
* Administrators are trained to handle the token authentication process securely.
* Technological infrastructure meets current security standards.

## 4.2 Constraints

* The token authentication process may be affected by system downtimes or regulatory changes.
* Secure devices and internet access are required for administrators and users.

# 5. Process Flow - Process and Procedures

## **5.1. Initiating Token-Based Authentication:**

* **Claimant/Subscriber Action:**
  + The claimant/subscriber provides the generated token along with their demographic data as part of the authentication process.
* **Verifier Action:**
  + The verifier initiates the authentication request by starting online and requesting a token for authentication.
* **Output:** Authentication request with token and demographic data is initiated.

## **5.2. Sending Token and Credentials for Authentication:**

* **System Action (Public Network Systems - Client):**
  + The client system sends the token, demographic details, and verifier credentials to the server for authentication.
  + These details are masked and encrypted to ensure secure transmission.
* **Output:** Secure transmission of token, demographic data, and verifier credentials to the server.

## **5.3. Verifying Credentials and Token:**

* **System Action (Server):**
  + The server first checks if the verifier credentials match the expected credentials.
  + It then verifies if the token and demographic data provided by the claimant match the records in the database.
  + The server checks if the token is still valid and not expired.
* **Output:** Credentials and token verification process is conducted.

## **5.4. Handling Successful Authentication:**

* **System Action (Server):**
  + If the verifier credentials match, and both the token and demographic data are validated, and the token is not expired:
    - The server stores the successful token authentication status in the claimant's UIN (Unique Identification Number) account with appropriate encryption and hashing for security.
    - A notification generator sends confirmation notifications to the claimant and verifier, including the timestamp of the successful authentication.
* **Output:** Successful authentication is logged, and notifications are sent.

## **5.5. Handling Unsuccessful Authentication Attempts:**

* **System Action (Server):**
  + If the verifier credentials do not match, the token or demographic data is incorrect, or the token is expired:
    - The system increments the exception and error handling counter.
    - The system allows up to three retry attempts for authentication.
    - If retry limits are exceeded, the system terminates the process, records the timestamp of the failed attempt, and sends a notification of the unsuccessful authentication.
* **Output:** Unsuccessful authentication attempts are logged, retry attempts are tracked, and notifications are sent.

## **5.6. Logging and Status Update:**

* **System Action (Server):**
  + The server logs all activities during the token-based authentication process, including both successful and failed attempts.
  + Status updates are stored in the system logs for auditing, compliance, and monitoring.
* **Output:** Detailed logs and status updates are created for compliance and security monitoring.

## **5.7. Process Termination:**

* **System Action (Server):**
  + The process terminates after successful authentication, enabling secure access to the system or service for the claimant.
  + In case of authentication failure, the process terminates with security measures in place, such as logging the attempt and sending out failure notifications.
* **Output:** Process concludes with either successful access granted or termination due to security concerns.

# 6. Visualisation

A screenshot of a computer

Description automatically generated

Please refer to the [GitHub](https://github.com/alan-turing-institute/Standard-Operating-Procedures-for-Digital-Identity-Systems) repository for further information.