PAR Quick-Start Guide

Multi-Authentication Broker and Signing Process

Version	Datum	Person	Vorgenommene Anpassungen/Bemerkungen
1.0	08.06.2024	Adrian Dilita	Initial version
1.1	10.06.2024	Ingmar Schmidt	Redaction of content
1.2	25.06.2025	Ingmar Schmidt	Inclusion of "AUD" parameter for ETSI signing interface's URL

Disclaimer: ***The guide below will cover only the basic flow and necessary variables to obtain a signature. For information about parameters not covered in this guide, please consult the full integration guide available for download in the partner section of our homepage (behind partner login).***

The PAR end-to-end flow consists of these steps:

1. PAR Request
2. Authentication
3. Identification (one-time)
4. Token Generation
5. Signature Generation

Requirements:

To start a signing, an OIDC client/secret and a certificate/key are required.

For demo purposes, these can be provided by the Support team (servicedesk.ict@swisscom.com).

The certificate and key will need to be added to your preferred API platform (ex. Postman, Insomnia) for these hosts:

- auth-trustservices.mtls-scapp.swisscom.com
- sign-trustservices.mtls-scapp.swisscom.com
- ais.swisscom.com

A viable authentication method will also be needed -- **for the scope of this guide, MobileID will be used.**

URLs/Endpoints:

https://auth.trustservices.swisscom.com	Browser host, usable **only with a token or with a redirect_uri** generated by a PAR request
https://auth-trustservices.mtls-scapp.swisscom.com	MTLS protected host, used for PAR/CIBA requests and token generation*
https://auth-trustservices.mtls-scapp.swisscom.com/api/auth/realms/broker/protocol/openid-connect/ext/par/request	PAR endpoint, used to generate a request_uri for the authentication step
https://auth-trustservices.mtls-scapp.swisscom.com/api/auth/realms/broker/protocol/openid-connect/token	Token endpoint, used to generate the SAD token for the signing endpoint
https://sign-trustservices.mtls-scapp.swisscom.com/etsi/standard/rdsc/v1/signatures/signDoc OR *(see JWT Token Response "AUD" parameter"* https://ais.swisscom.com/AIS-Server/etsi/standard/rdsc/v1/signatures/signDoc	MTLS protected signing endpoint, receives the SAD token and responds with a certificate

*For the purposes of this guide, only the PAR flow will be detailed

Steps for MAB E2E flow:

1 PAR Request

The parameters below must be replaced by the OIDC client details obtained from support and a MobileID-activated phone number.

Request Method	**POST**
Host	https://auth-trustservices.mtls-scapp.swisscom.com/api/auth/realms/broker/protocol/openid-connect/ext/par/request
Request format	*JSON*
Request body	{ "client_id": "**CLIENT_ID**", "client_secret": "**CLIENT_SECRET**", "login_hint": { "identifier": "**PHONE_NUMBER**", "namespace": "msisdn" }, "scope": "sign ident", "client_session_id": "test_session_id", "redirect_uri": " https://auth.trustservices.swisscom.com ", "claims": { "credentialID": "OnDemand-Advanced4 ", "documentDigests": [{ "hash": "sTOgwOm+474gFj0q0x1iSNspKqbcse4IeiqlDg/HWuI=", "label": "Leasing agreement Contract 1" }], "hashAlgorithmOID": "2.16.840.1.101.3.4.2.1" } }

On a successful request, MAB replies with a request_uri.

2 Authentication

To trigger an authentication page, an authentication URL must be constructed in url-encoded format:

client_id	OIDC client_id, obtained from support
request_uri	Generated by previous request
state	Unique UUID to identify the transaction, set by customer
nonce	Unique nonce, set by customer

A sample of an auth URL is available below:

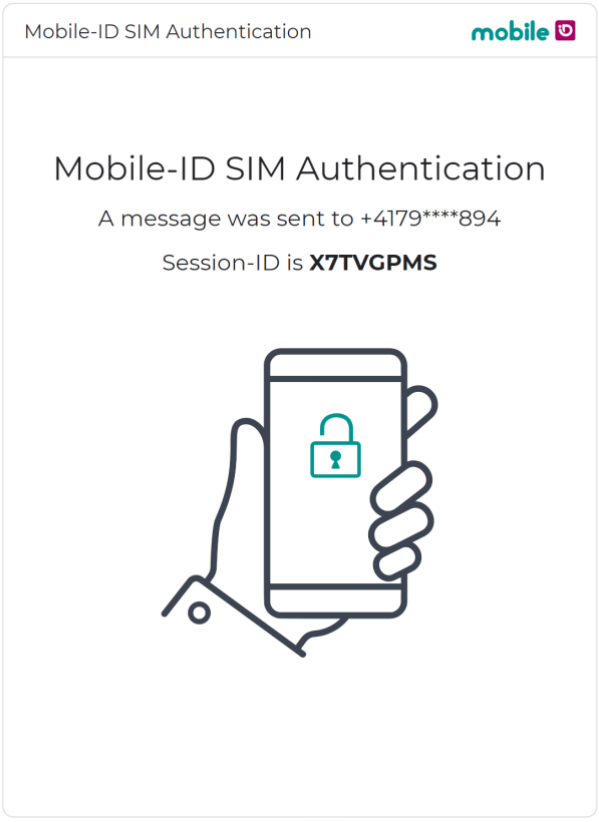
https://auth.trustservices.swisscom.com/auth/realms/broker/protocol/openid-connect/auth?client_id=**CLIENT_ID**&request_uri=**REQUEST_URI**&state=622c7ddb-cde8-4005-bed1-111111111111&nonce=1234-1234-12-34

! Note: the request_uri **must be url-encoded** in the final URL.

Example:

Unencoded (provided by PAR response)	**Encoded**
urn:ietf:params:oauth:request_uri:56ad856b-329d-4295-a44a-09a2d8be32c7	urn%3Aietf%3Aparams%3Aoauth%3Arequest_uri%3A56ad856b-329d-4295-a44a-09a2d8be32c7

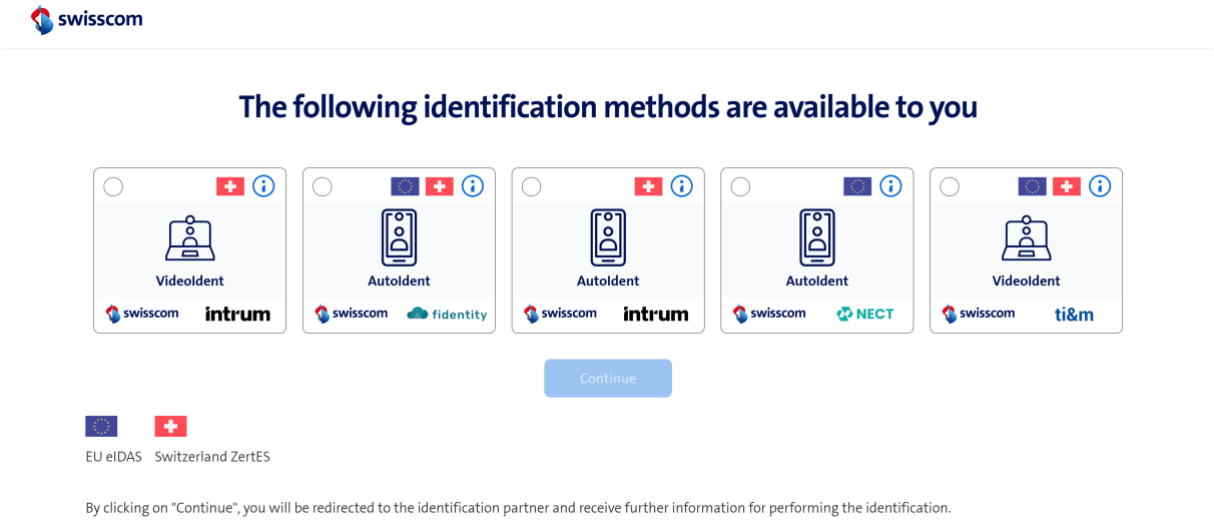
After constructing the authentication URL, access it via browser. A MobileID page will be displayed in the browser and your mobile device will be prompted to authenticate via MobileID:



Confirm the MobileID prompt on your mobile device.

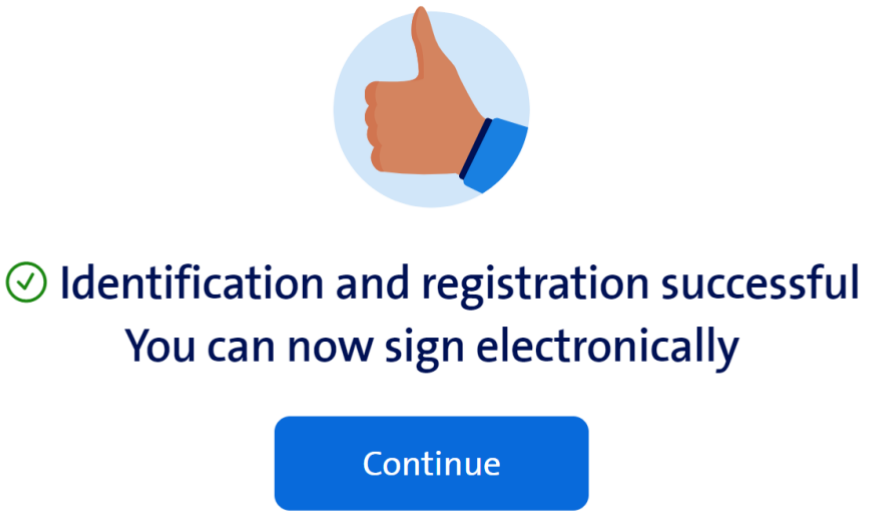
3 Identification

After confirmation, if you are not already onboarded in RAS, you will receive a list of identification partners to use. If you are already identified, skip to the end of the section to obtain the code.



Select any identification partner you want to use, accept the terms and conditions, and follow the identification process.

MAB will inform you once the identification process is successful:



Click **Continue**, and MAB will redirect to the landing page. In the URL you will receive the code needed for the signing token.

Example:

https://auth-trustservices.mtls-scapp.swisscom.com/en?state=622c7ddb-cde8-4005-bed1-111111111111&code=**252af8c8-7713-4923-86fa-a4b0920320b0**&refId=BW8UNH&identProcessId=TST-ZPFEGR-HF&identStatus=success&identInSigningFlow=true&identMethod=intrum-video

4 Token generation

The request will prompt MAB to respond with an SAD token, used in the ETSI signing interface* (see URL hint).

Request Method	**POST**	
Host	https://auth-trustservices.mtls-scapp.swisscom.com/api/auth/realms/broker/protocol/openid-connect/token	
Request format	*Form-URL-encoded*	
Request body	grant_type	authorization_code
	code	Obtained in step 3.2
	client_id	OIDC client ID, obtained from support
	client_secret	OIDC client secret, obtained from support

* The SAD token (JWT encoded) contains as response the parameter “aud” listing the right ETSI signing interface URL for your chosen use-case. You are required to extract the “aud” parameter to address the right signing interface with your signing request in step 5.

5 Signature Generation

The parameter in **red** must be replaced by the SAD token generated in the previous step.

Request Method	**POST**
Host	**Host URL is part of the JWT Token response, parameter “aud” containing this URL.**
Request format	*JSON*
Request body	{ "SAD": "**SAD_TOKEN**", "requestID": "8b6ed6f9-5bf8-42b2-bc8d-b242a35c94c8", "credentialID": "OnDemand-Advanced4", "profile": "http://uri.etsi.org/19432/v1.1.1#/creationprofile#", "signatureFormat": "P", "conformanceLevel": "AdES-B-LT", "documentDigests": { "hashAlgorithmOID": "2.16.840.1.101.3.4.2.1", "hashes": ["sTOgwOm+474gFj0q0x1iSNspKqbcse4IeiqlDg/HWuI="] } }

On a successful request, MAB will respond with the signature object, containing the document certificate.