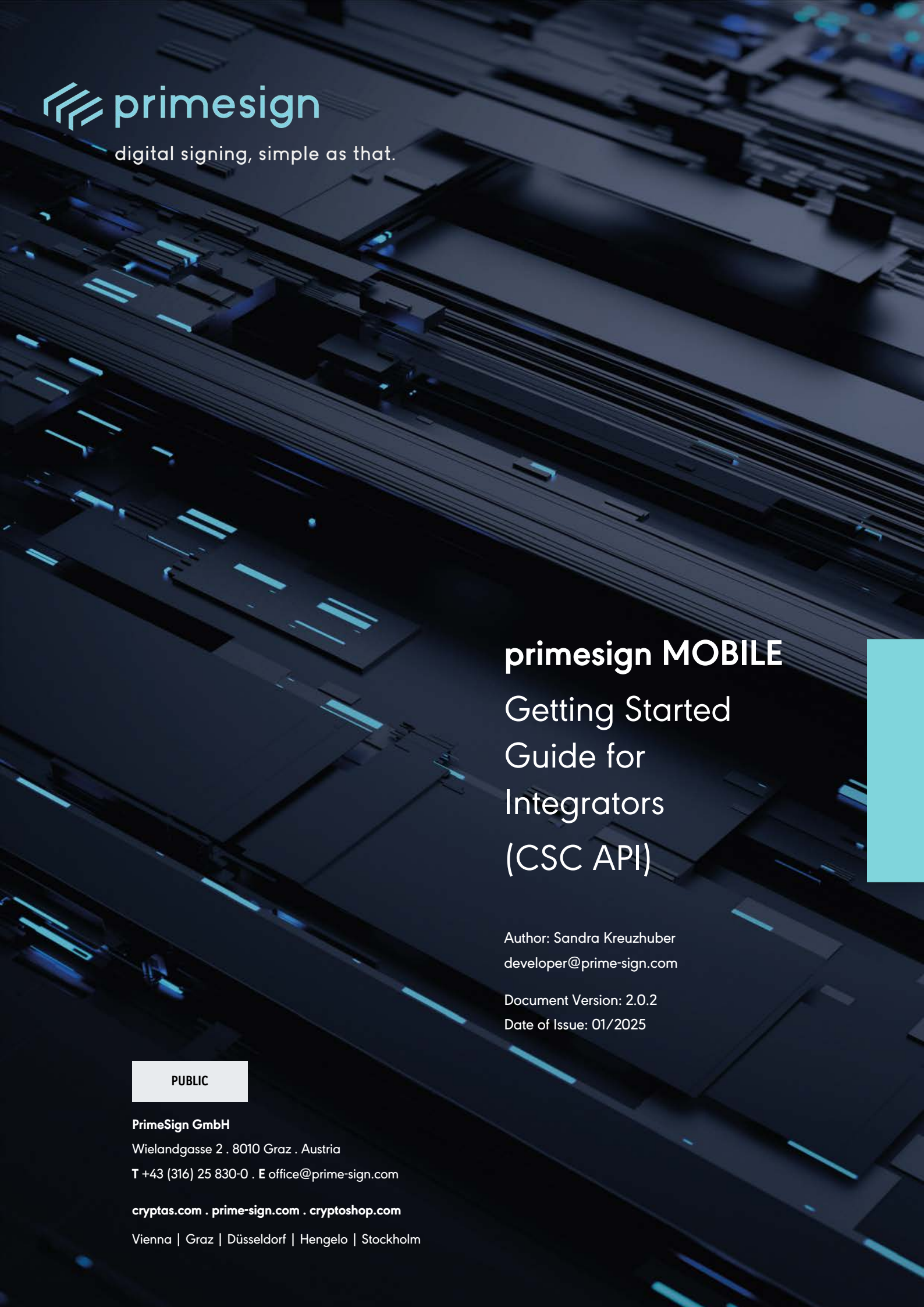


primesign

digital signing, simple as that.

primesign MOBILE

Getting Started Guide for Integrators (CSC API)

Author: Sandra Kreuzhuber
developer@prime-sign.com

Document Version: 2.0.2
Date of Issue: 01/2025

PUBLIC

PrimeSign GmbH
Wielandgasse 2 . 8010 Graz . Austria
T +43 (316) 25 830-0 . **E** office@prime-sign.com

cryptas.com . prime-sign.com . cryptoshop.com
Vienna | Graz | Düsseldorf | Hengelo | Stockholm

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. Document Information

primesign MOBILE allows the creation of qualified electronic signatures, both for natural and legal persons. To protect the security and interests of primesign MOBILE end users, primesign requires all integrators to fulfil certain conformance requirements and criteria and pass a conformance test.

This document serves provides guidelines on integrating primesign MOBILE and useful information for integrators.

1.1. Revision History

All changes to the document are tracked in the following history.

Date	Name	Type of Change	Version
24.04.2023	Sandra Kreuzhuber	Initial Version	1.0.0
03.07.2023	Sandra Kreuzhuber	Extension for Sign with eID	1.1.0
10.10.2023	Sandra Kreuzhuber	Extend Billing	1.1.1
19.04.2024	Sandra Kreuzhuber	Extension for public clients (native apps) and "Sign with eID"	2.0.0
13.08.2024	Sandra Kreuzhuber	Update account_token for public clients. Update document limit from 30 to 300. Add sequence diagram for pushed authorization.	2.0.1
30.01.2025	Sandra Kreuzhuber	Update Usage Agreement and Conformance Criteria	2.0.2

2. Service Overview

2.1. Feature Overview

primesign offers qualified signing via its remote signing solution primesign MOBILE[1]. With primesign MOBILE, documents are signed with an eIDAS-compliant qualified signature - conveniently and legally binding.

Customers can either use "Sign with eID" to sign with primesign MOBILE instantly by using their national eID[2] (no prior user registration with primesign required, signing with primesign MOBILE one-time certificates) or have a personal signing certificate issued promptly within minutes via a fully remote onboarding service. Here, the identity verification of applicants is done online via video or based on an existing electronic identity (eID), including, e.g., ID Austria and the Austrian mobile phone signature (Handy-Signatur).

For integrators, the same API implementation allows to

- **"Sign with eID"** to sign with primesign MOBILE instantly by using a national eID (this service is also called primesign WRAPTOR) and
- **sign with a personal primesign MOBILE signing certificate** (typically valid for 5 years).

[1] https://www.prime-sign.com/products/primesign-mobile
[2] https://www.prime-sign.com/sign-with-eid

2.2. Billing

The following tables summarize the two available billing options:

- Using primesign MOBILE with primesign ENTERPRISE ACCOUNT
- Or using primesign MOBILE without primesign ENTERPRISE ACCOUNT

2.2.1. With primesign ENTERPRISE ACCOUNT

primesign ENTERPRISE ACCOUNT	Yearly fee per integrator or organization
primesign MOBILE Transactions	Additionally: Billing per signature, volume discounts apply. Billed automatically to the primesign ENTERPRISE ACCOUNT. Monthly billing.
primesign MOBILE BASIC VOUCHER	Optional: One-time fee per individual user. Registrations remain valid for 5 years. *Note: individual onboarding and registration is only necessary for those users who do not sign instantly with an eID.*

Table 1: Billing with primesign ENTERPRISE ACCOUNT

Use primesign MOBILE in combination with one or more primesign ENTERPRISE ACCOUNTs to benefit from all features of primesign signatures.

- **Supports signing with a personal primesign MOBILE certificate or "Sign with eID" to sign instantly by using a national eID (primesign WRAPTOR).**
- Every signature transaction will be billed monthly against a primesign ENTERPRISE ACCOUNT. Allows to either use one primesign ENTERPRISE ACCOUNT per integrator and/or to have separate primesign ENTERPRISE ACCOUNTS per end customer (organization). The monthly invoice is always sent to the organization associated with the primesign ENTERPRISE ACCOUNT (e.g. the billing contact). Contact primesign to get a primesign ENTERPRISE ACCOUNT.

- Each primesign ENTERPRISE ACCOUNT is assigned a unique accountId. The integrator transmits the accountId of a primesign ENTERPRISE ACCOUNT in API requests as part of a so-called *account_token*.
- **For users that do not sign with their national eID:** primesign MOBILE vouchers are required. A voucher entitles for the issuance of a personal primesign MOBILE certificate (valid for 5 years). primesign offers several voucher types.
    - If a primesign ENTERPRISE ACCOUNT is used for billing the signing transactions, we recommend primesign MOBILE Basic vouchers. primesign MOBILE Basic vouchers can either be purchased directly by the user via online shop of primesign/CRYPTAS or the integrator can resell primesign MOBILE vouchers (reseller agreement required). primesign MOBILE Basic vouchers cover only the initial registration fee for primesign MOBILE; signing transactions are billed to the primesign ENTERPRISE ACCOUNT.
    - For users that sign frequently (e.g. executives), we recommend primesign MOBILE FLAT vouchers. primesign MOBILE FLAT is paid yearly and covers registration and unlimited signing transactions. Mixed-Use is possible, e.g. to use primesign MOBILE FLAT for frequent signers and primesign MOBILE BASIC for occasional signers.

2.2.2. Without primesign ENTERPRISE ACCOUNT

The simplest form to use primesign MOBILE:

- No transaction costs for integrators.
- Users (or their organizations) pay for the registration and the signing transactions. Therefore, users (or their organizations) purchase primesign MOBILE vouchers. A voucher entitles for the issuance of a personal primesign MOBILE certificate (valid for 5 years). Vouchers can either be purchased by users directly via online shop of primesign/CRYPTAS or the integrator can resell primesign MOBILE vouchers (reseller agreement required).
- **"Sign with eID" is NOT supported.** However, national eIDs can be used during registration for the personal primesign MOBILE certificate.

Overview of available primesign MOBILE vouchers, that can also be used without primesign ENTERPRISE ACCOUNT:

primesign MOBILE FLAT VOUCHER	Yearly fee per individual user. Includes registration and unlimited signature transactions.
primesign MOBILE 10 VOUCHER	One-time fee per individual user. Includes registration and 10 signature transactions.
primesign MOBILE 5 VOUCHER	One-time fee per individual user. Includes registration and 5 signature transactions.
primesign MOBILE Pay-per-Use VOUCHER	One-time fee per individual user. Includes registration. Additionally monthly billing of signing transactions. Customer receives invoice.

Table 2: Billing without primesign ENTERPRISE ACCOUNT

2.3. Service Information & Online Documentation

primesign provides up-to-date service documentation regarding registration, certificate lifecycle (suspension, revocation) as well as contact information to support and maintenance at the following URLs:

- English version: https://primesign.cryptas.com/hubfs/PDFs_primesign/primesign_Service_Informationen_EN.pdf
- German version: https://primesign.cryptas.com/hubfs/PDFs_primesign/primesign_Service_Informationen_DE.pdf

Register at https://status.prime-sign.com to receive automatic notifications about maintenance windows, updates and the current service status for primesign MOBILE.

Furthermore, the following URLs provide support options and a selection of frequently asked questions about primesign MOBILE from an end user perspective:

- English version: https://primesign.cryptas.com/en/primesign-support
- German version: https://primesign.cryptas.com/de/primesign-support

2.4. Contact & Support

2.4.1. Technical Contact

During initial integration phase, we prefer direct contact. Contact address for technical questions regarding the integration is developer@prime-sign.com.

As soon as the integration is successful and the production use starts, the integrator will receive the contact information to primesign support.

2.4.2. Commercial Contact

For commercial questions please contact Matthias Pankert (matthias.pankert@cryptas.com) or alternatively Thomas Rössler (thomas.roessler@prime-sign.com).

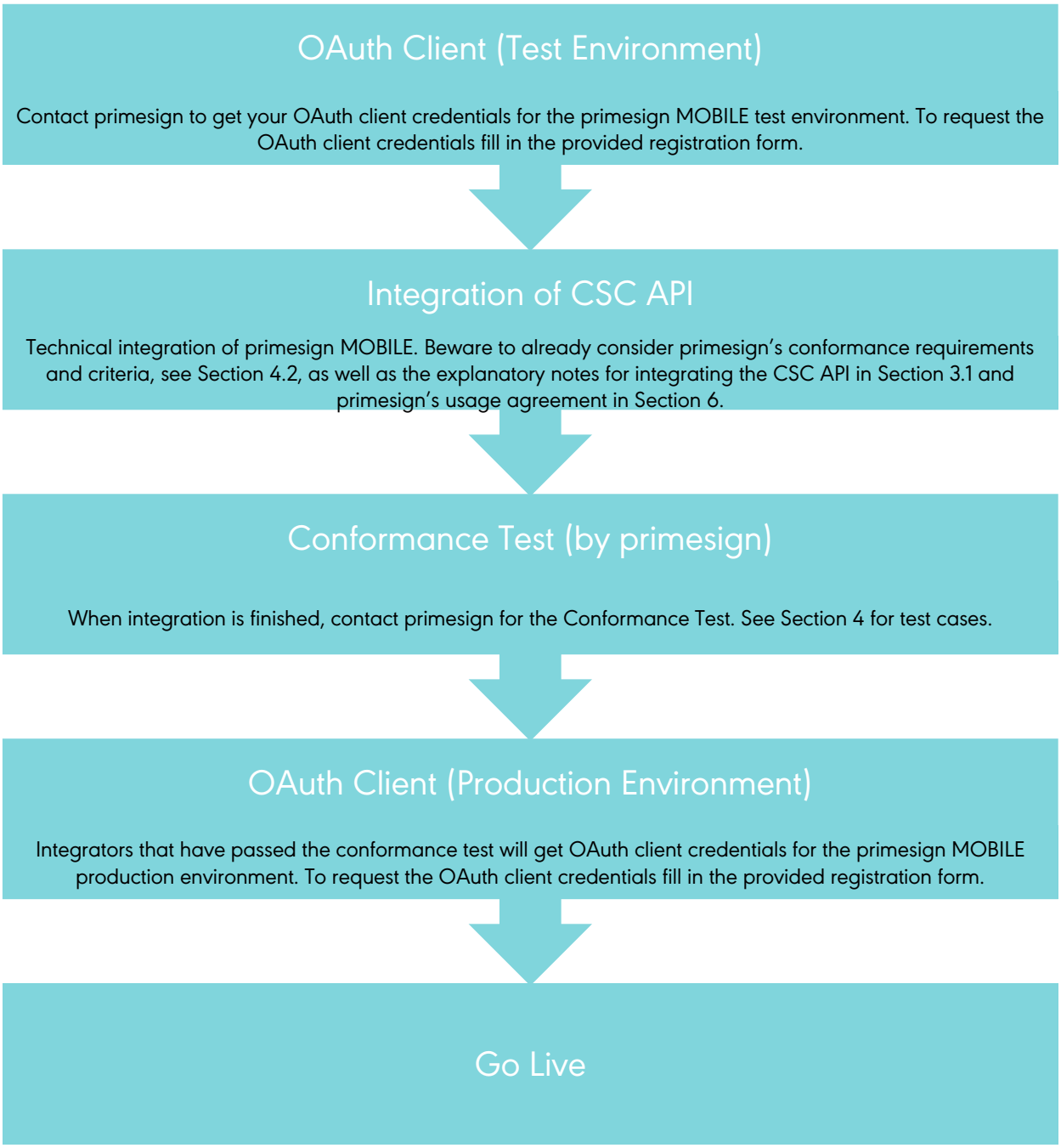
2.4.3. Support Contact

PREMIUM SLA: premiumsupport.cryptas.com

Alternative: basicsupport@cryptas.com

3. Integration Procedure

The following graph provides a simplified process of the single steps required for integrating primesign MOBILE:



4. Technical Integration

4.1. Explanatory Notes to CSC API

4.1.1. Documentation

primesign MOBILE is an eIDAS compliant remote signing service. primesign MOBILE is member of the Cloud Signature Consortium and implements the CSC API. For a detailed API documentation please refer to the official CSC API documentation (1).

Furthermore, attached you will find a Postman Collection including example requests that reflect all API calls, that are implemented by primesign.

4.1.2. CSC API Version

The integrator should implement the CSC API V1. primesign implements CSC API Version 1.0.4.0[3]. Version 0 is deprecated.

4.1.3. Supported CSC API Methods

The API method `/csc/v1/info` provides an up-to-date list of all supported API methods.

Currently the following CSC API Methods are supported:

- `/csc/v1/info`, see (1) Section 11.1
- `/credentials/list`, see (1) Section 11.4
- `/credentials/info`, see (1) Section 11.5
- `/signatures/signHash`, (1) see Section 11.9

For authentication solely authType `oauth2code` is supported. The url to the primesign IDENTITY PROVIDER (primesign's authentication server that handles OAuth authorization) is provided in the response parameter oauth2. For OAuth handling the following API methods are supported:

- `/oauth2/authorize`, (1) see Section 8.3.2
- `/oauth2/token`, see (1) Section 8.3.3
- `/oauth2/revoke`, see (1) Section 8.3.4

If the integrator supports Pushed Authorization, the following API methods are used:

[3] https://cloudsignatureconsortium.org/wp-content/uploads/2020/01/CSC_API_V1_1.0.4.0.pdf

- `realms/qs/protocol/openid-connect/ext/par/request`, see (2), Section 2.1

All authentication related API methods can also be found at the following URL:

https://id.prime-sign.com/realms/qs/.well-known/openid-configuration

4.1.4. Sequence Diagrams

For integrators, the same API implementation allows to

- "Sign with eID" to sign with primesign MOBILE instantly by using a national eID (this service is also called primesign WRAPTOR) and
- sign with a so-called *persistent* personal primesign MOBILE signing certificate (typically valid for 5 years).

However, there are two different ways for performing the OAuth 2 authorization flow:

- **Standard OAuth 2.0 authorization**, where the payload of an OAuth 2.0 authorization request is transmitted via the browser.
- **OAuth 2.0 authorization via Pushed Authorization Requests** [RFC9126[4]] which allows confidential clients to push the payload of an OAuth 2.0 authorization request to the authorization server via a direct request (server-to-server communication) and provides them with a request URI that is used as reference to the data in a subsequent call to the authorization endpoint. Beware: Pushed Authorization Requests can only be used with confidential clients, see section 4.2.2.1

The following diagram shows the full process flow for integrating primesign MOBILE. This example shows a signature with primesign MOBILE account. For users that use "Sign with eID" steps 18-20 differ. When signing with eID the entire user authentication is performed in steps 4-6 and may differ depending on the used eID.

[4] https://datatracker.ietf.org/doc/html/rfc9126

4.1.4.1. Signing with standard OAuth 2.0 authorization:

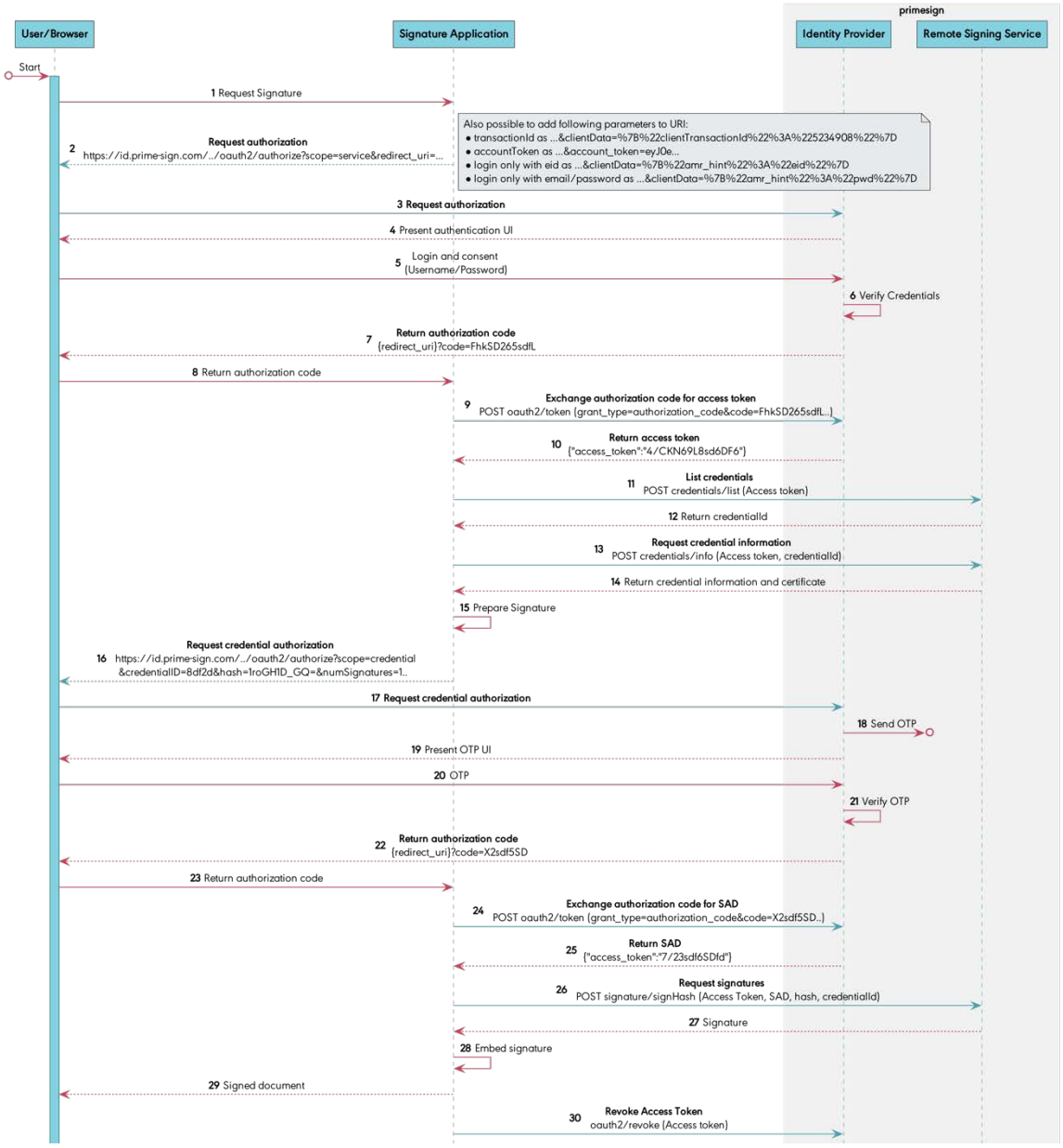
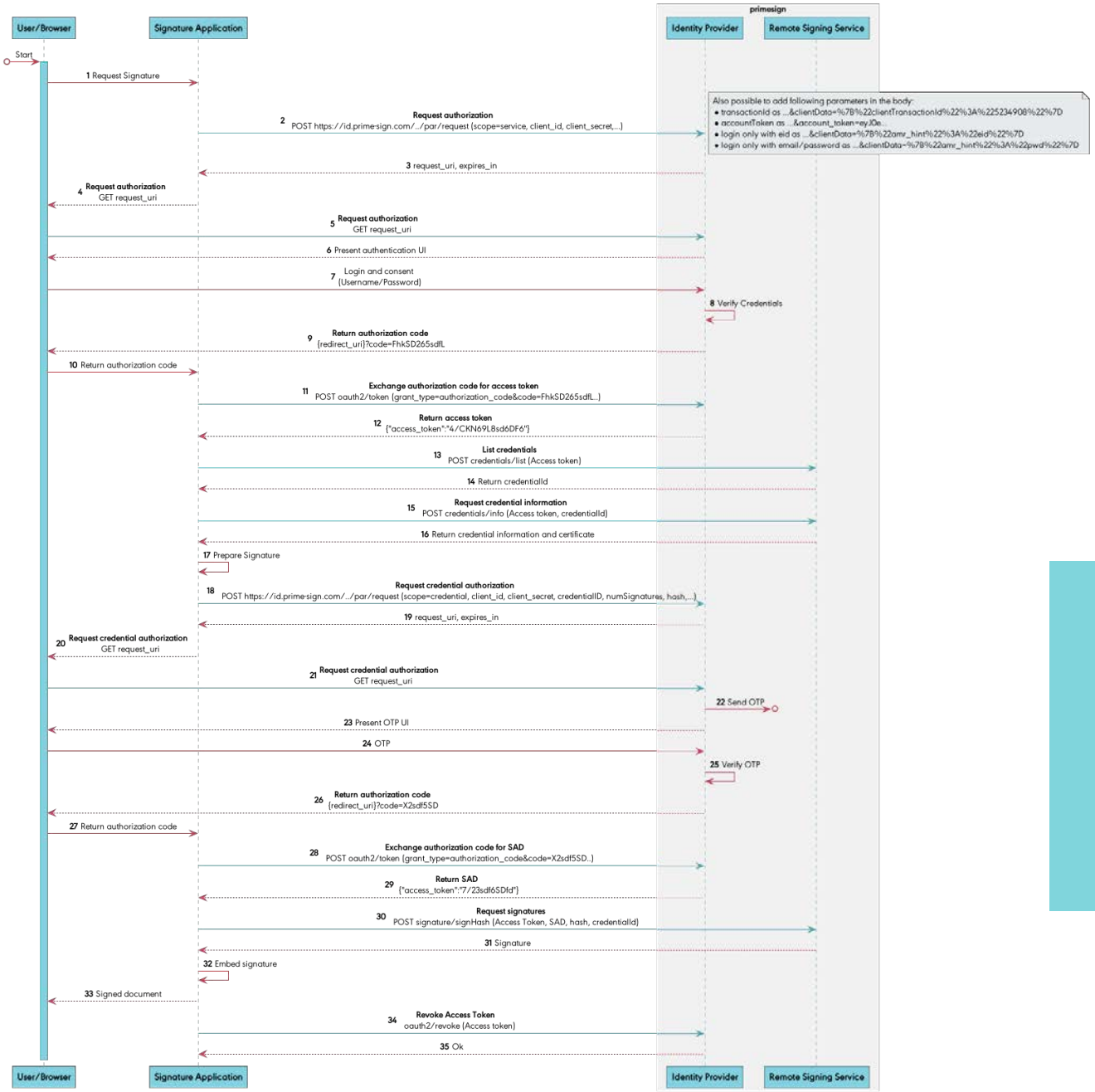


Figure 1: Sequence Diagram Signing with Standard OAuth 2.0 Authorization

4.1.4.2. Signing with OAuth 2.0 Pushed Authorization Requests:



4.1.5. signatures/signHash API method

4.1.5.1. Signature Algorithm

Signature keys created via primesign OnBoarding Service are ECC keys only. The integrator retrieves the supported signature algorithm for a specific credentials via `/credentials/info` in response attribute key/algo. Make sure to always use one of these supported signature algorithms for parameter signAlgo in API method `/signatures/signHash`.

4.1.5.2. Hash Value

Make sure to encode the hash value correctly before passing it to the API methods `oauth2/authorize` and `/signatures/signHash`. In accordance with the CSC specification, the hash value must be base64-**url** encoded when passing it to the `oauth2/authorize` endpoint (or when using Pushed Authorization: the `/par/request` endpoint). Whereas, for the API method `/signatures/signHash`, the hash value must be base64 encoded.

Example for oauth2/authorize (credential authorization):

```
GET /oauth2/authorize? response_type=code&
client_id=<OAuth_client_id>&
redirect_uri=<OAuth_redirect_uri>&
scope=credential&
credentialID=<credentialId>&
numSignatures=1&
hash=1roGH1D_jmf1FMgv0nQTELLdNHOtRaEK_7yhp_eBAGQ=
```

Example for `signatures/signHash`:

```
POST /csc/v1/signatures/signHash
Authorization: Bearer eyJhbGciOiJSUzI1NiIsInR5cCIgO…
{
  "SAD" : "_ eyLpskInJhbGciOiJSUzI..",
  "hash" : ["1roGH1D/jmf1FMgv0nQTELLdNHOtRaEK/7yhp/eBAGQ="],
  "hashAlgo" : "2.16.840.1.101.3.4.2.1",
  "signAlgo" : "0.4.0.127.0.7.1.1.4.1",
  "credentialID" : "<credentialID>"
}
```

4.1.6. Bulk Signing

primesign supports bulk signing, where multiple hashes can be signed in one single step, i.e. with just one signature authorization. Billing per signature transaction, regardless of whether 1 or 10 documents are signed in one step.

Limit for bulk signing: 300 hashes

Hint: Use `/credentials/info` method to retrieve the current limit for bulk signing.

Furthermore, beware of following limitations:

- Browsers have length limitations for URLs opened in the browser. The URL length limitations of standard browsers allow to send between 30 and 50 hashes via `oauth2/authorize` requests. Beware of the length limitations of the used browsers and/or webviews and implement your own limit for bulk signing accordingly.
- If your application wants to sign more hashes, you can also implement OAuth 2.0 Pushed Authorization Requests, see section 4.1.4.2. Beware that Pushed Authorization Requests can only be used by confidential clients! See section 4.2.2.1 for an overview of confidential and public clients.

4.1.7. Notes on Credential Validity

primesign permits signing only with enabled and valid credentials. Currently, each user is only allowed to have one credential. primesign issues credentials (certificates) for a validity of typically 5 years. Revoked or expired credentials and the associated user will be deleted automatically. Suspended credentials will be marked as disabled while suspended. After a suspension was lifted, the credential is enabled again.

4.1.8. primesign ENTERPRISE ACCOUNT

Optional

When using primesign MOBILE with a primesign ENTRERPRISE ACCOUNT, every signature transaction will be billed monthly against this primesign ENTERPRISE ACCOUNT. You can either use one primesign ENTERPRISE ACCOUNT per integrator and/or have separate primesign ENTERPRISE ACCOUNTS per end customer (organization). The monthly invoice is always sent to the organization associated with the primesign ENTERPRISE ACCOUNT.

Each primesign ENTERPRISE ACCOUNT is assigned a unique accountId. The integrator transmits the accountId of a primesign ENTERPRISE ACCOUNT in API requests as part of a so-called

`account_token`. The `account_token` is a JSON Web Token (JWT), that includes the accountId of a primesign ENTERPRISE ACCOUNT.

For using `account_token` consider:

- An `account_token` must be passed in the service authorization call, when calling `oauth2/authorize` for the first time (or when using Pushed Authorization: when calling `/par/request` the first time). Passing an `account_token` in the credential authorization is optional.
- The `account_token` is a JWT consisting of header, payload and signature. The accountId of the primesign ENTERPRISE ACCOUNT is added as "sub" attribute in die JWT Payload. See Section 8.3.1 of the CSC API specification for the format of the account_token.
- The JWT Payload also includes the issuing time of the JWT. `account_token` are valid for 2 minutes (2 minutes from the issuing time given in "iat").
- The JWT_signature required to generate the `account_token` SHALL be calculated with the HMAC function, using the SHA256 hash of the client secret.

Example account_token for confidential clients (decoded):

```
{
  "header": {
    "typ": "JWT",
    "alg": "HS256"
  },
  "payload": {
    "iss": "<SOME IDENTIFIER OF YOUR COMPANY OR SOFTWARE>",
    "sub": ""<ACCOUNT ID OF PRIMESIGN ENTERPRISE ACCOUNT>",
    "iat": 1721220528,
    "azp": ""<YOUR CLIENT ID>",
    "jti": "3c6492e7-3df2-416f-b7d5-389870a287d6"
  },
  "signature": "pn9gAhogNcK2mPZrMlV7JbuLX85tTUPTHYZicJf8ZIc"
}
```

Example account_token for public clients (decoded):

Public clients (e.g. desktop apps) do not have client secrets. Therefore, public clients must send unsigned account_tokens. primesign accepts unsigned account_tokens only from public clients. Confidential clients must sign the transmitted account_token.

```
{
  "header": {
    "alg": "none",
    "typ": "JWT"
  },
  "payload": {
    "sub": "<ACCOUNT ID OF PRIMESIGN ENTERPRISE ACCOUNT>",
    "jti": "3c6492e7-3df2-416f-b7d5-389870a287d6",
    "iss": "<SOME IDENTIFIER OF YOUR COMPANY OR SOFTWARE>",
    "azp": "<YOUR CLIENT ID>",
    "iat": 1718872680
  },
  "signature": ""
}
```

Please refer to the CSC specification for the technical implementation and to developer@prime-sign.com for further information to the corresponding billing configuration.

4.1.9. Transaction Identifier

Optional

primesign recommends passing a `clientTransactionId` for both `/oauth2/authorize` requests. The `clientTransactionId` is used for support and billing (included in transaction reports). The `clientTransactionId` is passed in JSON format in the `clientData`.

- Example: `/oauth2/authorize?response_type=code&clientData=%7B"clientTransactionId"%3A"asdf"%7D…`
- Allowed characters: a-z A-Z 0-9 _ @ : + . -
- Maximum length: 200 characters

4.1.10. Further Best Practices

1. Make sure to display the service name as well as the logo of primesign MOBILE without distortions/pixelation etc.
2. Make sure to handle cancel actions (“Cancel” button in primesign MOBILE UI).
3. Do not integrate primesign MOBILE in an iFrame. Using WebViews is not encouraged. WebViews are only allowed after explicit approval by primesign.
4. Revoke the access token (from the service authorization) after signing using the API method `/oauth2/revoke`.

5. primesign recommends adhering to OAuth2 Best Practices[5] , e.g. to use PKCE. PKCE is mandatory for public clients.
6. Review the conformance requirements and criteria in section 5.2 before the conformance test

4.2. API Access

4.2.1. Service Endpoints

These are the base URLs for primesign MOBILE:

Test: https://qs.primesign-test.com

Production: https://qs.prime-sign.com

primesign forbids load tests with services provided by primesign (includes both test and production environment). Exceptions only with prior authorization from primesign.

4.2.2. OAuth Client

User authentication in primesign MOBILE is implemented via OAuth. Therefore, each application (client) requires client credentials (clientId, *for confidential clients only:* client secret) for the primesign IDENTITY PROVIDER. The client credentials are issued by primesign. Different client credentials for test and production systems are required.

When issuing client credentials, primesign requires a list of redirect URIs for the client.

To get client credentials for test and production please fill in the attached registration form and send by e-mail to developer@prime-sign.com. primesign recommends to make use of established security mechanisms supporting the security of OAuth authorization, such as PKCE. For native apps / public clients the use of PKCE is mandatory.

primesign will send you the client credentials (clientId, *for confidential clients only:* client secret) in encrypted form. Further instructions will follow.

Client credentials for production will only be issued after the conformance test was passed.

4.2.2.1. Confidential Clients (default)

[5] https://datatracker.ietf.org/doc/html/draft-ietf-oauth-security-topics

By default, primesign creates confidential clients for the integrating applications. A confidential client is characterized as follows:

- Confidential clients are applications that are able to securely authenticate with the primesign IDENTITY PROVIDER
- Confidential clients will get client credentials consisting of clientId and client secret
- Confidential clients can hold credentials in a secure way without exposing them to unauthorized parties. Therefore, they require a trusted backend server to store the client secret.

The following restrictions apply for redirect URIs:

- primesign supports simple wildcards, e.g. "https://example.com/*".
- Redirect URIs must be issued under a public toplevel domain (e.g. no *.local or lo-calhost URIs).
- Regular Expressions are supported, but not recommended.
- It is possible to supply redirect URIs with different domains.
- The integrator confirms control over the domain under which the redirect URIs are assigned.

The integrator is obliged to prevent unauthorized access to credentials of the primesign MOBILE testing and production environment. This applies in particular to the client secret. In case unauthorized access to the client secret is detected, contact primesign immediately to reset the client secret.

4.2.2.2. Public Clients / Native Apps

primesign creates public clients exclusively for applications that have no possibility to securely store a client secret, such as apps without corresponding backend server. All other applications must use confidential clients.

Public clients are characterized as follows:

- Client-side applications such as mobile apps or native desktop apps or client-side web applications (single page web apps without corresponding backend server)
- Public clients are unable to use client secrets, as there is no way to securely deploy client secrets (shipping a client secret would require to ship the secret in the binary distribution of the application which is easy to decompile and extract).

- When issuing public clients, primesign creates a clientId for the application. There is no client secret for public clients. Important: As there is no client secret, public clients cannot use OAuth 2.0 Pushed Authorization Requests, see section 4.1.4.2.
- Public clients MUST use the OAuth2 Authorization Code Flow. Using PKCE is mandatory for public clients. This prevents authorizations codes from being used by a different application than the one that started the authorization.
- Applications should launch the system browser for OAuth authorization. Using WebViews is not encouraged. WebViews are only allowed after explicit approval by primesign. See https://www.oauth.com/oauth2-servers/oauth-native-apps/use-system-browser/ for more information.
- See https://www.oauth.com/oauth2-servers/oauth-native-apps/ and RFC 8252 for more recommendations on OAuth 2.0 for Native Apps.

The following restrictions apply for redirect URIs:

- ***Recommended:*** Some platforms (Android, iOS, Windows apps) allow to register custom URL schemes, so when the mobile browser or some other mobile apps opens the custom URL (for example the redirect URI), the native app opens automatically. Example: com.example.app:/oauth2redirect/example-provider. The integrator confirms control over the domain under which the claimed custom URIs are assigned.
- ***Alternatively, if custom URL schemes are not available:***
    a. When using localhost URLs (loopback redirect URIs): URLs such as http://127.0.0.1:[port]/path] are allowed. Arbitrary and thus also randomly assigned port numbers are allowed.
    b. When using claimed HTTPS URIs: The integrator confirms control over the domain under which the claimed redirect URIs are assigned.
- primesign supports simple wildcards, e.g. "https://app.example.com/some-path/*".
- Regular Expressions are supported, but not recommended.

4.3. Testing & Test Users

For initial integration and during future developments primesign provides a public test environment (https://qs.primesign-test.com).

The test environment allows to test signing with:

- "Sign with eID" to sign with primesign MOBILE instantly by using a national eID (this service is also called primesign WRAPTOR) and
- sign with a personal primesign MOBILE signing certificate (typically valid for 5 years).

Our test environment issues certificates from our test hierarchy only (no qualified certificates!). Therefore, beware that these signatures may be displayed as "untrusted" by various signature verification tools.

When using primesign MOBILE with primesign ENTERPRISE ACCOUNT, primesign recommends to transmit account_token already during integration with the primesign test environment. Contact primesign to get one or multiple accountIds for primesign ENTERPRISE ACCOUNTs within the test environment (no costs apply in the test environment).

4.3.1. Testing with national eIDs ("Sign with eID")

If enabled for the test OAuth client, users can use a (test) eID for signing.

Country	**Allowed Identities with primesign test environment**
DE	primesign test environment accepts German Test IDs only. The easiest way for testing is to install AusweisApp on Desktop and enable the developer settings. Furthermore, enable the option "Aktiviere den internen Kartensimulator" to authenticate without a physical ID card. See https://www.ausweisapp.bund.de/ausweisapp2/help/1.22/de/Windows/settings-developer.html
AT	primesign test environment accepts "real" ID Austria accounts and test accounts. For testing, public test identities issued by the provider of ID Austria can be used. See https://eid.egiz.gv.at/anbindung/testidentitaeten/vordefinierte-testidentitaeten/ for a list of test IDs.

Beware, if you want to use "Sign with eID" with the test environment, the application must transmit an `account_token`.

4.3.2. Testing with primesign MOBILE accounts

For testing signature creation with persistent primesign MOBILE certificates, test accounts are necessary for the integrator. Each test account requires a registration code (VOUCHER CODE). A limited number of registration codes are provided by primesign for free. Inform primesign how many test accounts are required.

To register a test account, visit the primesign OnBoarding Service (https://onboarding.primesign-test.com). Fill in the registration data and make sure to use a valid e-mail address and mobile

phone number. For identification you can either use your eID (e.g. ID Austria/Handy-Signatur) or use the test system of our video identification partner. When video identification is used, skip the actual video call by entering the magic TAN "123456".

After successful identification, we issue test certificates from our test hierarchy (no qualified certificates!). Only ECC keys are issued.

On request, primesign also issues accountIds for primesign ENTERPRISE ACCOUNTs within the test environment (no costs apply in the test environment).

4.3.3. Automatic Testing

For signature creation with persistent primesign MOBILE certificates, a password and SMS-TAN is required. However, within the test environment primesign can mark specific accounts to work with our magic TAN „123456". This is especially useful during development, when multiple developers share one test account and for integration tests, to allow for periodic automatic testing. Contact primesign to configure one of your existing test accounts to work with the magic TAN.

These test accounts with magic TAN are only available in the primesign MOBILE test environment.

Signing with primesign MOBILE instantly by using a national eID (primesign WRAPTOR) cannot be tested automatically, as a national eID is required for signing.

4.3.4. Production Environment

Integrators will receive a limited number of registration codes (VOUCHER CODES) for our production environment upon request and for free. These registration codes can be used to register for primesign MOBILE via primesign OnBoarding Service (https://onboarding.prime-sign.com). When registering, the users identify via video boarding or eID (e.g., ID Austria/Austrian mobile phone signature). During the onboarding procedure, a real qualified certificate is issued. Beware, that this certificate is bound to a specific person (the person who was identified via video or eID/Austrian mobile phone signature) and can be used to create legally binding qualified electronic signatures. Thus, beware not to share credentials to production accounts for primesign MOBILE (i.e. no sharing between developers).

If signing with primesign MOBILE instantly by using a national eID (primesign WRAPTOR) is enabled for the integrator, users can use their own national eID for signing.

4.4. UI Integration

4.4.1. Logo & Brand Names

The `/csc/v1/info` API method provides the URL to the primesign logo. Contact primesign in case you need other logos (other sizes, resolutions etc.).

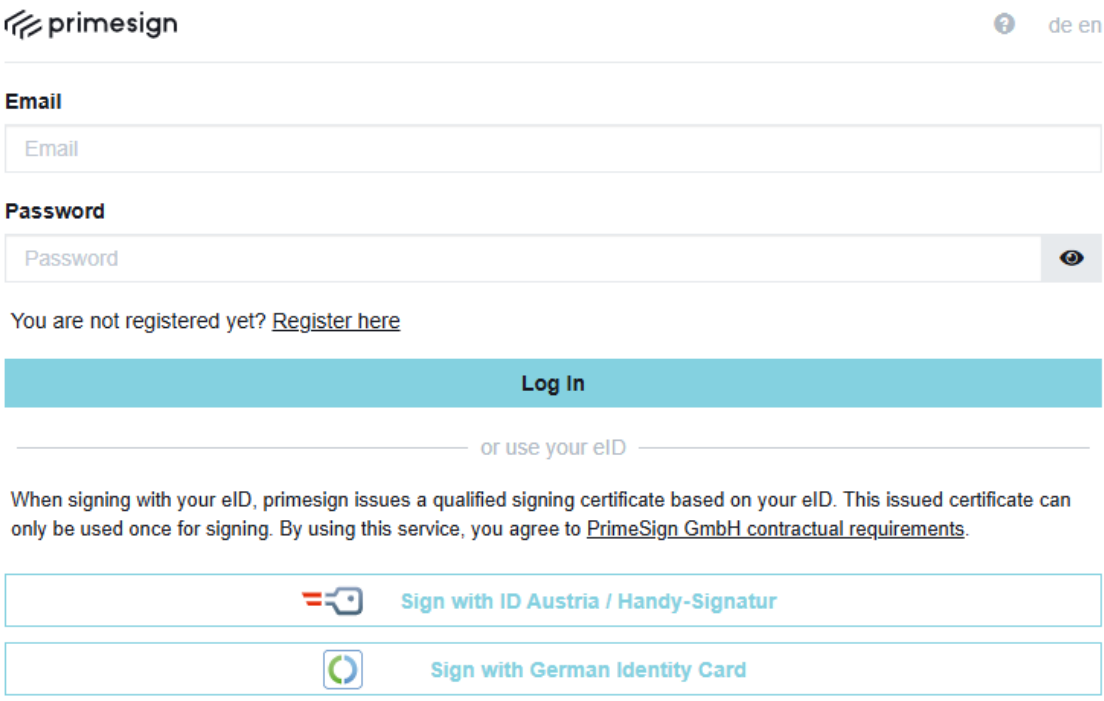
Beware to use the correct brand names:

- When speaking of our company in general or when the users should choose their signature providers: primesign
- When referring to our remote signing service: primesign MOBILE
- When referring to our trust center (issuer of qualified certificates): primesign TRUST CENTER

4.4.2. Signing with eID

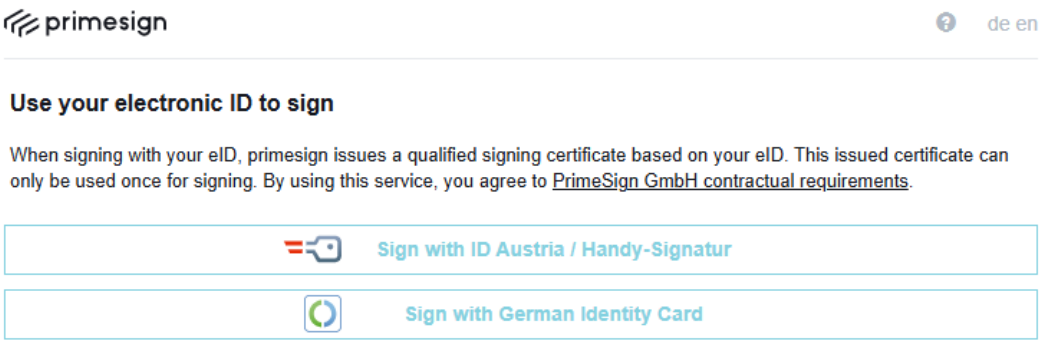
Optional.

If signing with primesign MOBILE instantly by using a national eID (primesign WRAPTOR) is enabled for the integrator, the option to signing with eID is displayed below the username and password login dialog. See screenshot below.

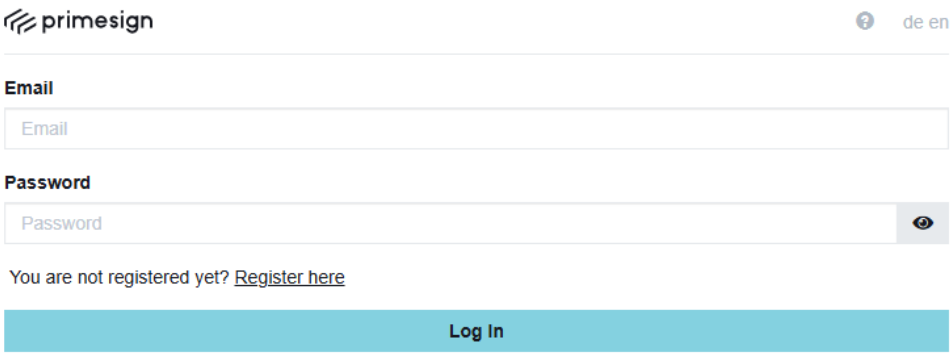


primesign also allows to hide the option to log in with email and password from the login dialog. This may be used if an integrator offers signing with eID only.

Therefore, primesign allows passing an `amr_hint` in `/oauth2/authorize` requests. The `amr_hint` with value "eid" hides the option to log in with email and password from the login dialog. See screenshot below.



The `amr_hint` "pwd" hides the option to sign with primesign MOBILE using a national eID (primesign WRAPTOR) from the login dialog. See screenshot below



The `amr_hint` is passed in JSON format in the `clientData`.

- Example: `/oauth2/authorize?scope=service&clientData=%7B"amr_hint"%3A"eid"%7D...`
- Supported value: "eid", "pwd".

5. Conformance Test

primesign will perform a conformance test in the testing environment of the integrator. In case direct access for primesign testers is not possible, a conformance test can be scheduled to take place in a joint call with screensharing.

5.1. Test Cases

The following test cases will be checked:

- **Test Case 1: Signing a single document with primesign MOBILE**
  Signing a single document with primesign MOBILE.
- **Test Case 2: Re-signing a single document with primesign MOBILE**
  Performing two signatures within the lifetime of the same user session (15 minutes).
- **Test Case 3: Signing multiple documents with primesign MOBILE**
  Optional; only if the integrator supports bulk signing.
- **Test Case 4: Signing a single document with primesign MOBILE on a mobile device**
  Optional; Will be tested if integrator provides a mobile app or specific web app for mobile devices that differs from the desktop web app.
- **Test Case 5: Signing a single document by using a national eID (primesign WRAPTOR)**
  Optional; Only tested if signing with primesign MOBILE instantly by using a national eID (no prior user registration with primesign required, signing with primesign MOBILE one-time certificates) should be enabled for the integrator.
- **Test Case 6: Signing multiple documents by using a national eID (primesign WRAPTOR)**
  Optional; depends on the integrator. Only tested if bulk signing with a national eID is supported.

5.2. Conformance and Test Criteria

The following sections summarize the conformance requirements and criteria as tested during the conformance test. The integrator represents and warrants that at the time of the conformance documented in the Conformance Protocol the criteria listed below are fulfilled and shall remain fulfilled at all times during the use of the primesign product/services in production environment. Any deviation from this will result in the immediate loss of the right to use the primesign services. All other rights and remedies that PrimeSign may have at law shall remain reserved.

5.2.1. User Interface Integration & Branding

Requirements/Criteria	Status

The application makes sufficiently transparent, that the user is about to electronically sign a document (or a list of documents).	REQUIRED
The application enables the user to view the document(s) to be signed.	REQUIRED
The name "primesign" is written correctly. For example, use the name "primesign" when user choose their signature provider.	REQUIRED
The primesign logo is embedded nicely. (No distortions, no pixelation)	REQUIRED
The application resigns on using an iFrame to display the primesign MOBILE UI.	REQUIRED
If a WebView is used: Using WebViews is not encouraged. WebViews are only allowed after explicit approval by primesign.	REQUIRED
The application handles the Cancel-Action ("Cancel" button in primesign MOBILE UI) properly.	REQUIRED

Table 1: User Interface Integration & Branding

5.2.2. OAuth Authorization Requests

Requirements/Criteria	**Status**
The application may only fetch a service access token with the intention that a signature is executed immediately afterwards.	REQUIRED
The application fetches a new service access token in case the user intends to create another signature within a short period of time (session lifetime).	REQUIRED
Revoke the service access token after signing.	REQUIRED
Only relevant if signatures should be billed via primesign ENTERPRISE ACCOUNT: The accountId is correctly encoded as account_token. The signature transaction can be billed to the purchased primesign ENTERPRISE ACCOUNT.	OPTIONAL
primesign recommends passing a clientTransactionId for both oauth2/authorize requests. The clientTransactionId facilitates support enquiries and is included within transaction reports. Format requirements: max: 200 characters, allowed characters: a-zA-Z0-9_@;+.-	OPTIONAL
The application uses PKCE.	OPTIONAL | REQUIRED for public clients

Table 2: OAuth Authorization Requests

5.2.3. CSC API Requests

Requirements/Criteria	Status
The application uses CSC API Version v1.	REQUIRED
The application creates ECDSA signatures.	REQUIRED

Table 3: CSC API Requests

5.2.4. Signature Verification

Requirements/Criteria	Status
The resulting signed PDF document is verified correctly with Adobe Acrobat PDF reader.	REQUIRED
The resulting signed PDF document is verified correctly with DSS[6].	REQUIRED
The resulting signed PDF document is verified correctly with RTR Tool[7].	REQUIRED
The resulting signed PDF document is LTV enabled.	OPTIONAL

Table 4: Signature Verification

[6] https://ec.europa.eu/digital-building-blocks/DSS/webapp-demo/validation
[7] https://www.rtr.at/TKP/was_wir_tun/vertrauensdienste/Signatur/signaturpruefung/Pruefung.de.html

6. Usage Agreement

The integrator warrants to comply with all required requirements and criteria communicated to the integrator in advance established in primesign MOBILE Getting Started Guide for Integrators (CSC API)[10]. Any deviation from these required requirements and criteria will result in the immediate loss of the right to use the primesign services. Additionally, the integrator shall indemnify and hold primesign harmless for all damages and costs, including those of third parties resulting from or related to non-compliance with the required requirements of Section 3.2.

primesign may check the continued compliance with the conformance criteria at any time and shall receive all reasonable support and evidence from the integrator upon request. The costs for this shall be borne by the integrator.

This conformance statement is valid for an indefinite period and until revoked by primesign.

Furthermore, the integrator agrees to the following terms and conditions:

- The integrator adheres to IT security best practices and guidelines on secure software development.
- The integrator shall make it sufficiently transparent, that the end user is about to sign a document or multiple documents. The integrator shall enable end users to view each document(s) to be signed before signing. Apart from changes that are required for preparing the document for signature creation (e.g. adding a signature widget, adding the signing certificate and revocation information), no changes are permitted.
- The integrator is obliged to prevent unauthorized access to credentials of the primesign MOBILE testing and production environment. For confidential clients this applies in particular to the client secret. In case unauthorized access to the client secret is detected, integrator shall contact primesign immediately to reset the client secret. primesign does not issue client secrets to public clients.
- The integrator is obligated to inform primesign in advance in case of non-trivial changes (irrespective if these are made directly by the integrator or its business partners) within the process flow, the way the PDF signatures are created or UI integration that are covered by this conformance test. This especially applies to any changes that impact any of the criteria listed in section **Error! Reference source not found.**. In case of changes, a new conformance test is required before rollout by the integrator or its business partners. primesign reserves the right to immediately deactivate API access to primesign MOBILE if

[10] primesign MOBILE Getting Started Guide for Integrators (CSC API): https://primesign.cryptas.com/de/hash-signing-api

the requirements and criteria of Section 3.2 are not met. For the avoidance of doubt, the integrator is entitled to make changes to the process flow with the OAuth test client only. Such changes may be implemented with the productive OAuth client after having undergone and passed a new conformance test as described in the Conformance Protocol.

- primesign may register several redirect URIs of the integrator. The integrator confirms control over the domain under which the redirect URIs are assigned. Exceptions only apply to localhost URLs (loopback redirect URIs).
- The integrator shall not execute load tests with services provided by primesign (includes both test and production environment). Exceptions may apply only upon prior clarification with primesign.
- primesign recommends integrators to subscribe for maintenance and update announcements on the primesign MOBILE product via https://status.prime-sign.com/. primesign announces regular software updates for its services 2 weeks in advance. New product versions are then available on primesign's testing infrastructure for 2 weeks. primesign advises integrators to always test their primesign MOBILE integration after an update is announced.

7. References

1. **Consortium, Cloud Signature.** [Online] https://cloudsignatureconsortium.org/resources/download-api-specifications/.