

AGOV

Identity Provider Interface

Classification	Not classified
Status	Published
Project Leader	Philipp Dasen
Version	1.7
Date	13. August 2025
Contracting Authority	Peppino Giarritta
Author/Authors	H. Burger (Adnovum)
Distribution Cantons	DVS, BK, BIT

Table of Contents

1	Introduction	3
1.1	Change History	3
2	AGOV Usage Scenarios.....	4
2.1	Overview.....	4
2.2	Account Management.....	5
2.2.1	Account Management during authentication	5
2.2.2	AGOV me direct link for registration.....	7
2.2.3	AGOV accounts with higher AGOVaq.....	7
2.2.4	Configuration per eGov application	7
3	General requirements	10
3.1	System-time.....	10
3.2	HTTP protocol layer.....	10
3.3	Algorithms and key material for signing	10
3.3.1	SAML artifacts.....	10
3.3.2	OIDC artifacts	10
3.4	Algorithms and key material for encryption	11
4	SAML interface	12
4.1	References	12
4.1.1	Specifications.....	12
4.1.2	Namespaces	13
4.2	Definetime.....	14
4.2.1	eGov-Application Metadata.....	14
4.3	Runtime	16
4.3.1	Overview and supported bindings.....	16
4.3.2	SAML Entity ID and Endpoints.....	18
4.3.3	SAML AuthnRequest.....	18
4.3.4	SAML Response	20
4.3.5	SAML Assertion	23
4.3.6	SAML Response and Assertion processing.....	28
5	OIDC Interface	29
5.1	References	29
5.1.1	Specifications.....	29
5.2	Definetime.....	30
5.2.1	Support of public clients	31
5.3	Runtime	31
5.3.1	Supported Flows	31
5.3.2	Usage of UserInfo endpoint	33
5.3.3	eGov Application initiated Logout.....	34
5.3.4	Endpoints	34
5.3.5	Authentication Request	35
5.3.6	AGOV IdP Authenticates the User	37
5.3.7	Token Request.....	38
5.3.8	AGOV IdP processes the Token Request.....	39
5.3.9	Response and ID Token processing	44
6	Appendix.....	45
6.1	Attribute lengths.....	45
6.2	Sample messages	46
6.2.1	SAML.....	46
6.2.2	OIDC.....	51

1 Introduction

The present document contains the detailed description of the two interfaces provided by AGOV Identity Provider (IdP) to [e-government applications](#) (eGov Application).

It will later be extended to cover also the Interface between AGOV and federated e-government Identity Providers (eGov IdP) planned in a future release of AGOV.

eGov Applications and eGov IdPs build together the participants of AGOV.

The present document has a double function:

- It serves as base for the configuration of any e-government application which wants to use AGOV to authenticate its users.
- It serves as base for the configuration and implementation of the AGOV Identity Provider interface (IdP). AGOV IdP will be tested against this interface specification.

Note: The key words "**MUST**", "**MUST NOT**", "**REQUIRED**", "**SHALL**", "**SHALL NOT**", "**SHOULD**", "**SHOULD NOT**", "**RECOMMENDED**", "**MAY**", and "**OPTIONAL**" in the rest of this document are to be interpreted as described in [RFC 2119](#).

1.1 Change History

Version	Date	Author	Description
0.6	20.10.2023	H. Burger	Verification processes MUST always be triggered by relyingparty
1.0	27.10.2023	H. Burger, A. Aeschlimann	Miscellaneous updates and corrections
1.1	23.02.2024	H. Burger	Updated handling of addresses of AGOV account holders
1.2	15.05.2024	H. Burger	Clarification of the error codes returned by the IdP, if the user denies/postpones a necessary identityverification to reach the requested AGOVaq level
1.3	01.08.2024	H. Burger	Added appendix with field sizes, miscellaneous updates and corrections
1.4	11.09.2024	H. Burger	OIDC: enable end_session_endpoint for eGov Application initiated logouts
1.5	11.12.2024	A. Aeschlimann	Change email field length from 300 to 255, according to client request Sample assertion corrected (no additionalAddress or c/o field)
1.6	07.03.2025	H. Burger	OIDC: claim "address.formatted" will not contain the user's name anymore
1.7	13.08.2025	H. Burger	introduction of conversationId attribute / claim

2.2 Account Management

Users create and manage their accounts on their own. AGOV provides the AGOV me user portal for that purpose.

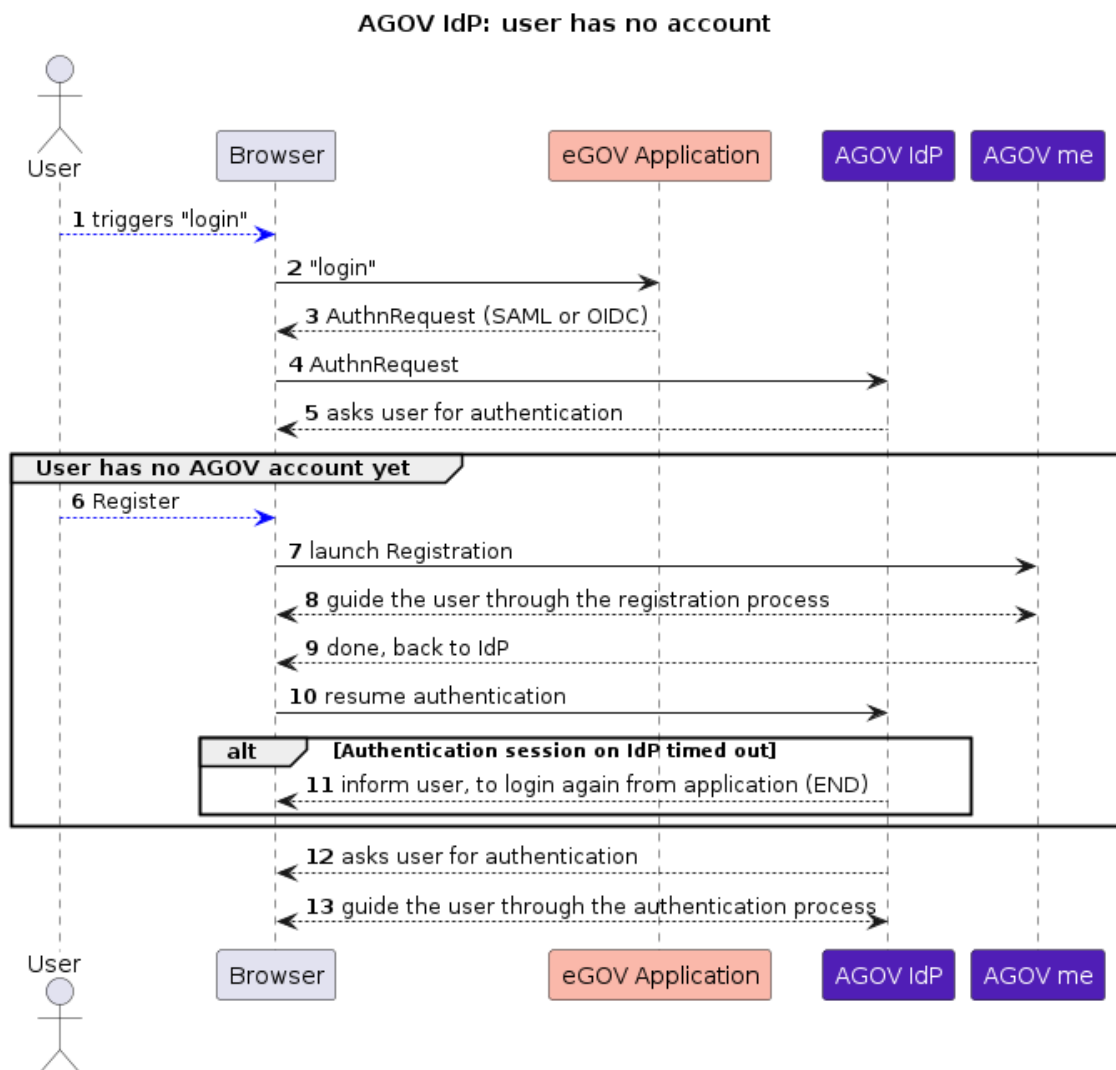
There are basically two ways an application can use to invite their users to set up an AGOV account:

- "on the fly" registration / identity verification during an authentication
- direct link to AGOV me (only for a basic account, without identity verification)

2.2.1 Account Management during authentication

i If the user registers or upgrades his account during an authentication, then there is no guarantee that the application will get a response to the authentication request. The authentication process might time out, thus the IdP loses the context and can't answer the request anymore. In that case, the user is invited to close the browser and start again with a login from the application.

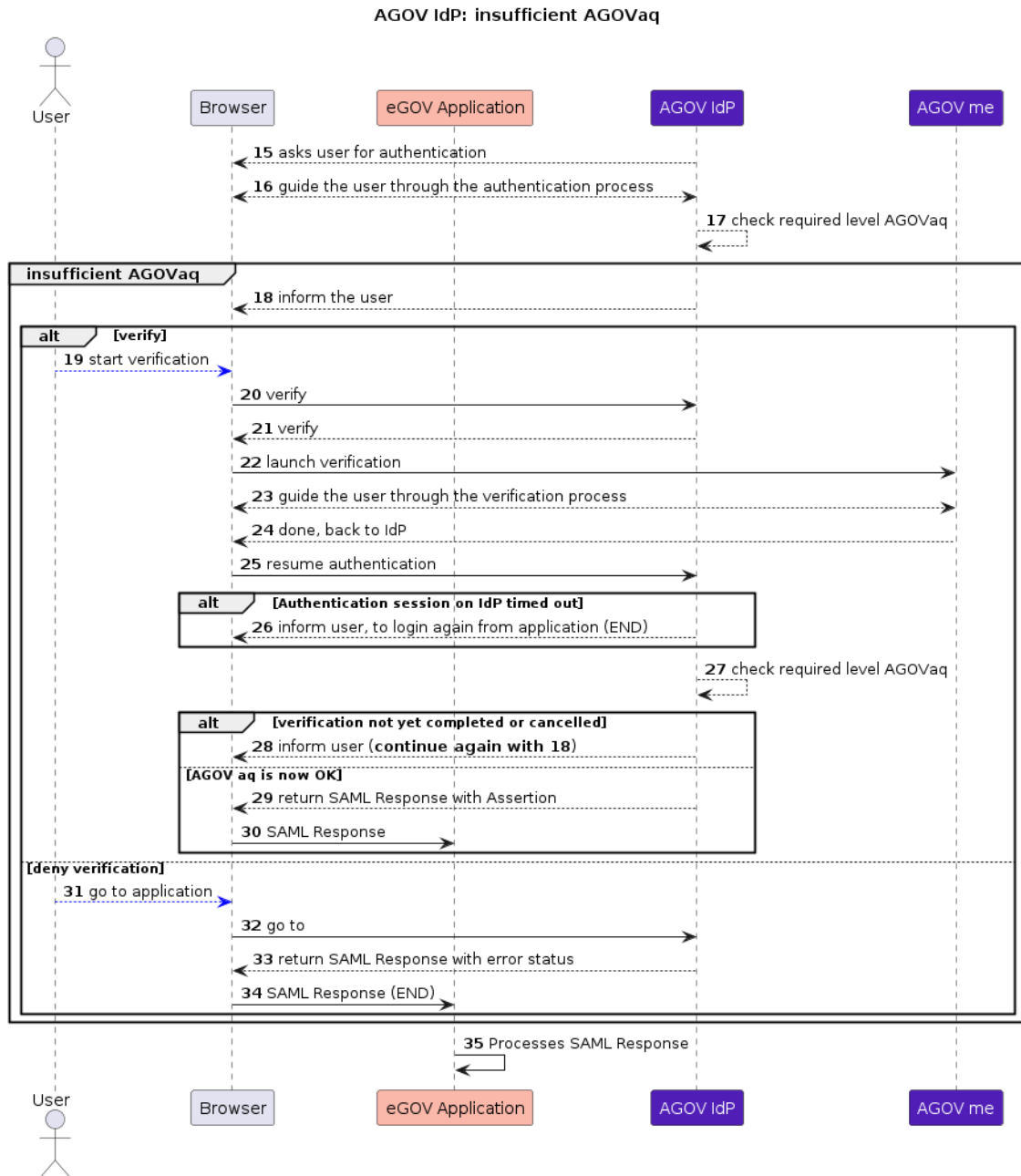
The user has the possibility to register an account during an authentication request, if he has no account yet, as shown on the diagram below:



The IdP redirects the user's browser to AGOV me for that purpose. After registration is done, AGOV me redirects the browser back to AGOV IdP to resume the authentication.

It might happen that the IdP lost the session meanwhile.

After a successful authentication, the AGOV IdP checks whether the user's account meets the required assurance quality level. If not, the user has the possibility, to start the verification process:



The user can decide to go directly back to the application. In that case, the AGOV IdP will return a SAML Response with an error status to the application. Otherwise the user will be guided through the verification process by AGOV me.

Notes:

- If the user registers a new account during an authentication which requires a higher AGOVaq, he is invited to start the verification process immediately after account registration.

- The following error codes are returned, if the user denies/postpones the necessary identity verification:
 - SAML: `urn:oasis:names:tc:SAML:2.0:status:NoAuthnContext`
 - OIDC: `access_denied`

2.2.2 AGOV me direct link for registration

eGOV applications can use the following Registration URL in their application, support portals, documentation to direct the users to AGOV me and invite them to set up an initial AGOV account:

URL	1.0
Registration URL	<code>https://www.me.agov.admin.ch/registration/</code>

The AGOV me portal will guide the user through account creation.

Caution: No identity or address verification will take place. However, the newly created account - with self-declared values - will immediately be usable.

2.2.3 AGOV accounts with higher AGOVaq

eGov application must send an authentication request to AGOV IdP with a specific minimal required AGOVaq level to trigger the account upgrade of a user.

If the user has not yet an account, he will be invited to create one within the same flow.

2.2.4 Configuration per eGov application

During the registration of the eGov application with AGOV connect, the application has the possibility to define the following requirements

- default AGOVaq: this is the minimal AGOVaq that will be enforced by AGOV IdP during authentication. This setting can be overwritten in the authentication request.

AGOVaq Level is one of the following:

AGOVaq	semantic
100	The identity of the user is not verified
200	The postal address is verified by successful mail delivery
300	The identity of the user is verified by a strong identity verification
400	The identity of the user is verified by a strong identity verification and completed with the verified social security number ("AHVNummer")

Level 500 (i.e. LOA3 according to eCH-170 V2) cannot yet be requested for registration or upgrading. This will be achievable through the future Swiss E-ID or some federated eGov IdPs.

- data in the token:
 - address required: Shall the address be added to the token.
 - 📘 Note: this has only an effect for AGOV accounts with an AGOVaq higher or equal to 200 and must be combined by requesting such an AGOVaq
 - social security number required: Shall this number be added to the token as an additional identifier
 - 📘 Note: this has only an effect for AGOV accounts with an AGOVaq higher or equal to 400 and must be combined by requesting an AGOVaq of at least 300

- allowed methods, for AGOVaq 300 verification: possible are currently BmID (identification service of Swiss Post; physical verification of the user's ID document by the postman, or at a Swiss post office), Video (online verification of the user's ID by a human agent), or Counter (physical verification of the user's ID at a counter operated by the canton, or a municipality). AGOV me will suggest the user a verification method based on the allowed methods for the eGov application, the user's domicile, and the users ID document.

2.2.4.1 Handling of the address

The following table shows the behavior of AGOV IdP with respect to the address:

eGov Application requests		agov account AGOVaq level				
AGOV aq	addr*	100	200	300	300	400
100	no	✗	✗	✗	✗	✗
100	yes	✗	✗	✗	✗	✗
200	no	n/a	✗	✗	✗	✗
200	yes	n/a	✔	✔	✔	✔
300	no	n/a	n/a	✗	✗	✗
300	yes	n/a	n/a	✔	✔	✔
400	no	n/a	n/a	n/a	n/a	✗
400	yes	n/a	n/a	n/a	n/a	✔

*) as declared by the application in AGOV connect

The table reads as follows:

- an application declared the need for the address in AGOV connect and sends an authentication request requiring AGOVaq 200, then the application will get token **with** the address ✔ provided the user has an account with at least AGOVaq 200
- an application declared the need for the address in AGOV connect and sends an authentication request requiring only AGOVaq 100, then the application will get a token **without** address ✗ independently of the AGOVaq level of the account

Address verification is currently only supported for Swiss addresses. Foreign addresses are self-declared. The table below shows how the address is verified depending on the the domicile country, and AGOVaq level:

AGOVaq	Swiss address	Foreign address
100	n/a	n/a
200	code sent to the address	n/a
300 - BmID	code sent to the address	self-declared
300 - Video or Counter	domicile or location certified*	self-declared
400 - BmID	code sent to the address	self-declared
400 - Video or Counter	domicile or location certified*	self-declared

*) Usage of an address verification service by Swiss Post for Swiss addresses, where either the quality "domicile certified" or "certified" is required (additional information: [Address verification via the web service](#)).

2.2.4.2 Handling of the social security number


i The social security number is only supported for users having a Swiss social security number (a number beginning with 756).

The following table shows the behavior of the AGOV IdP with respect to the social security number (svnr):

eGov Application requests		agov account AGOVaq level				
AGOV aq	svnr*	100	200	300	300	400
100	no	✗	✗	✗	✗	✗
100	yes	✗	✗	✗	✗	✗
200	no	n/a	✗	✗	✗	✗
200	yes	n/a	✗	✗	✗	✗
300	no	n/a	n/a	✗	✗	✗
300	yes	n/a	n/a	✗	✗	✔
400	no	n/a	n/a	n/a	n/a	✗
400	yes	n/a	n/a	n/a	n/a	✔

*) as declared by the application in AGOV connect.

Important: By requesting the svnr number through AGOV connect, the relying party confirms that their organization is aware of and adheres to [SR 831.10 - Bundesgesetz vom 20. Dezember 1946 ü... | Fedlex](#). AGOV operation declines all responsibility should svnr's be communicated to organisations which are not entitled to receive them.

Note: the social security number is delivered in the token , as soon as at least AGOVaq 300 is requested in the authentication request and the application was configured to receive the svnr. This allows applications to benefit of that number for people working in Switzerland and allowing also accounts of foreign people at the same time, with the same assurance level except the social security number.

3 General requirements

3.1 System-time

Each participant of AGOV that issues SAML messages, consumes SAML messages or OIDC ID tokens **MUST** synchronize its time with an accepted time server, in order to validate the timing constraints on these artifacts.

3.2 HTTP protocol layer

All externally reachable services from AGOV are based on the HTTP protocol.

The endpoints provided by AGOV **MUST** only be accessible over TLS.

Endpoints made available by AGOV participants for the use by AGOV (such as SAML metadata endpoints of eGOV applications) **MUST** also be accessible over TLS.

The TLS version **MUST** be 1.2 or higher. The server certificates used to authenticate the https endpoints **MUST** be issued by a generally trusted certificate authority.

3.3 Algorithms and key material for signing

3.3.1 SAML artifacts

The following applies for SAML AuthnRequests, SAML Responses, SAML Assertions, and SAML Metadata

Signature generation and validation is governed by [\[XML-DSIG\]](#).

SAML messages **MUST** be signed with a X.509 compatible algorithm and key.

The KeyInfo element of the signed XML document **MUST** contain the X.509 certificates corresponding to the private key used to sign the message.

The certificate **MUST** be either issued by an official certificate authority (CA) trusted by AGOV for that purpose or may be a self-signed certificate.

The currently trusted official CAs are:

- [Swiss Government Regular CA 01](#) (issuing so called class C certificates)

If self-signed certificates are used, then they **MUST** meet the following criteria:

- The Key-Pair **MUST only** be used for digital signing
- The chosen algorithm **MUST** be one of the following
 - RSA PSS with a key size of at least 3072bits
 - ECDSA with a key size of at least 256bits
- The validity of the certificate **MUST** be less than or equal to 12 months
- The CN of the subject **MUST** be equal to the SAML EntityID

3.3.2 OIDC artifacts

The following applies for ID Tokens and keys used for the private_key_jwt authentication method.

Signature generation and validation is governed by [\[JWS\]](#) and [\[JWA\]](#).

The chosen algorithm **MUST** be one of the following:

- RSA PSS with a key size of at least 3072bits
- ECDSA with a key size of at least 256bits

The key **MUST** be rotated at least every 12 months.

3.4 Algorithms and key material for encryption

i Encryption is only used with the SAML protocol in AGOV.

AGOV **SHOULD** support encryption of SAML Assertions in the context of the http POST binding, i.e. if the SAML Assertion passes through the user agent of the end user.

Whenever supported by the participants, the SAML artifact binding **SHOULD** be used by the AGOV participants to transmit SAML Assertions.

Encryption is governed by [\[XML-ENC\]](#).

Detailed requirements will be added, if there is a participant in AGOV which requires encryption.

4 SAML interface

4.1 References

4.1.1 Specifications

[SAML-CORE]	OASIS Standard, Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0 – Errata Composite Working Draft 07, 8 September 2015. [Online] https://www.oasis-open.org/committees/download.php/56776/sstc-saml-core-errata-2.0-wd-07.pdf
[SAML-BIND]	OASIS Standard, Bindings for the OASIS Security Assertion Markup Language (SAML) V2.0 - Errata Composite Working Draft 06, 8 September 2015. [Online] https://www.oasis-open.org/committees/download.php/56779/sstc-saml-bindings-errata-2.0-wd-06.pdf
[SAML-META]	OASIS Standard, Metadata for the OASIS Security Assertion Markup Language (SAML) V2.0 – Errata Composite Working Draft 05, 8 September 2015. [Online] https://www.oasis-open.org/committees/download.php/56785/sstc-saml-metadata-errata-2.0-wd-05.pdf
[SAML-META-UI]	OASIS Standard, SAML V2.0 Metadata Extension for Login and Discovery User Interface Version 1.0, 24 October 2019 [Online] https://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-metadata-ui/v1.0/sstc-saml-metadata-ui-v1.0.html
[SAML-META-EXT]	OASIS Standard, SAML V2.0 Metadata Extension for Entity Attributes Version 1.0, 4 August 2009 [Online] http://docs.oasis-open.org/security/saml/Post2.0/sstc-metadata-attr-cs-01.pdf
[SAML-PROF]	OASIS Standard, Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0 – Errata Composite Working Draft 07, 8 September 2015. [Online] https://www.oasis-open.org/committees/download.php/56782/sstc-saml-profiles-errata-2.0-wd-07.pdf
[SCHEMA1]	H. S. Thompson et al. XML Schema Part 1: Structures World Wide Web Consortium Recommendation, May 2001. http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/

[XML-DSIG]	XML Signature Syntax and Processing (Second Edition) W3C Recommendation, 10 June 2008. http://www.w3.org/TR/xmlsig-core/
[XML-ENC]	XML Encryption Syntax and Processing Version 1.1 W3C Recommendation, 11 April 2013. https://www.w3.org/TR/xmlenc-core1/
[OWASP-SAML]	SAML Security Cheat Sheet https://cheatsheetseries.owasp.org/cheatsheets/SAML_Security_Cheat_Sheet.html

4.1.2 Namespaces

SAML relies on XML documents and schemas. In the documentation the following prefixes will be used:

Prefix	XML Namespace	Comments
ds	http://www.w3.org/2000/09/xmlsig#	This namespace is defined in the XML Signature Syntax and Processing specification [XML-DSIG].
md	urn:oasis:names:tc:SAML:2.0:metadata	SAML V2.0 metadata namespace defined in the SAML V2.0 metadata specification [SAML-META].
mdui	urn:oasis:names:tc:SAML:metadata:ui	This is the SAML V2.0 metadata extension namespace defined in the SAML Metadata Extensions for Login and Discovery User Interface [SAML-META-UI].
mdattr:	urn:oasis:names:tc:SAML:metadata:attribute	This is the namespace defined SAML Metadata Extension for Entity Attributes [SAML-META-EXT].
saml2	urn:oasis:names:tc:SAML:2.0:assertion	SAML V2.0 assertion namespace defined in the SAML 2.0 core specification [SAML-CORE].
saml2p	urn:oasis:names:tc:SAML:2.0:protocol	SAML V2.0 protocol namespace defined in the SAML 2.0 core specification [SAML-CORE].
xenc	http://www.w3.org/2001/04/xmlenc#	XML Encryption Syntax and Processing [XML-ENC].
xsd	http://www.w3.org/2001/XMLSchema	This namespace is defined in the W3C XML Schema specification [SCHEMA1]. In schema listings, this is the default namespace and no prefix is shown.

4.2 Definetime

eGov-Applications which want to use AGOV for authentication with SAML **MUST** register with AGOV IdP before using it. Registration will be done through the selfservice portal AGOV connect.

Registration will be based on so called meta data. In a first phase this data is entered by hand, but can be processed later on automatically by AGOV connect, provided that the metadata of the eGov-Application is publicly reachable.

4.2.1 eGov-Application Metadata

The following data is required to register an eGov-Application for the SAML interface (metadata XML tag `md:EntitiesDescriptor/md:EntityDescriptor`):

The requirements are formulated with reference to the corresponding element in the SAML metadata standard and its extensions. The same information can also manually be entered through AGOV connect while registering the application.

Element / Attribute	Description	Availability
[entityID]	The eGov-Application MUST register a globally unique <code>entityID</code> . This ID is used as Issuer for the requests sent to the AGOV IdP The <code>entityID</code> SHOULD be a URL allowing to access the entities metadata.	mandatory
n/a	If the <code>entityID</code> doesn't allow discovery of the metadata, then the eGov-Application should register the URL to access those metadata.	optional
SPSSODescriptor/AssertionConsumerService	The eGov-Application MUST register at least one <code>AssertionConsumerService</code> (ACS) endpoint. The <code>Binding</code> attribute MUST be either <code>urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST</code> or <code>urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Artifact</code> . The <code>ACS Location</code> attribute MUST be unique within all registered eGov-Applications.	mandatory
SPSSODescriptor/ KeyDescriptor[use='signing']	The eGov-Application MUST register at least one signing certificate with AGOV (see also key requirements above).	mandatory

Element / Attribute	Description	Availability
SPSSODescriptor/ KeyDescriptor[use='encryption']	<p>The eGov-Application MAY register an encryption certificate with AGOV (see also key requirements above).</p> <p>If an encryption key is registered then the assertion will be encrypted with that key for transmission over the HTTP-POST binding.</p>	optional
SPSSODescriptor/AttributeConsumingService/ RequestedAttribute	<p>The eGov-Application MAY indicate, that she requires the social security number or the address in the emitted assertions.</p> <p>The application does so by defining a default <code>AttributeConsumingService</code> element, and adding <code>http://schemas.agov.ch/ws/2023/05/identity/claims/socialSecurityNumber</code> or <code>http://schemas.agov.ch/ws/2024/02/identity/claims/address</code> in the list of <code>RequestedAttribute</code></p> <p>Other attributes can't be explicitly requested, the AGOV IdP MUST ignore them.</p>	optional
SPSSODescriptor/Extensions/ mdattr:EntityAttributes/Attribute	<p>The eGov-Application MUST define the minimal default quality of assurance by setting the <code>EntityAttribute</code> with the name "urn:oasis:names:tc:SAML:attribute:assurance-certification" to the correct value.</p> <p>The value must be one of:</p> <ul style="list-style-type: none"> <code>urn:qa.agov.ch:names:tc:ac:classes:100</code> <code>urn:qa.agov.ch:names:tc:ac:classes:200</code> <code>urn:qa.agov.ch:names:tc:ac:classes:300</code> <code>urn:qa.agov.ch:names:tc:ac:classes:400</code> <code>urn:qa.agov.ch:names:tc:ac:classes:500</code> 	mandatory
SPSSODescriptor/Extensions/ mdui:UIInfo/mdui:DisplayName	<p>The eGov-Application MAY define a name to be displayed on the Login Screen during user authentication.</p> <p>The name SHOULD be defined in the four languages de, fr, it, and en.</p>	optional

Element / Attribute	Description	Availability
SPSSODescriptor/Extensions/ mdui:UIInfo/mdui:Logo	The eGov-Application MAY define a logo to be displayed on the Login Screen during user authentication. It is possible to provide different Logos per language de, fr, it, rm, and en.	optional
.../<ds:Signature>	The eGov-Application's metadata MUST be digitally signed with a certificate identifying the authority responsible to publish the metadata.	mandatory

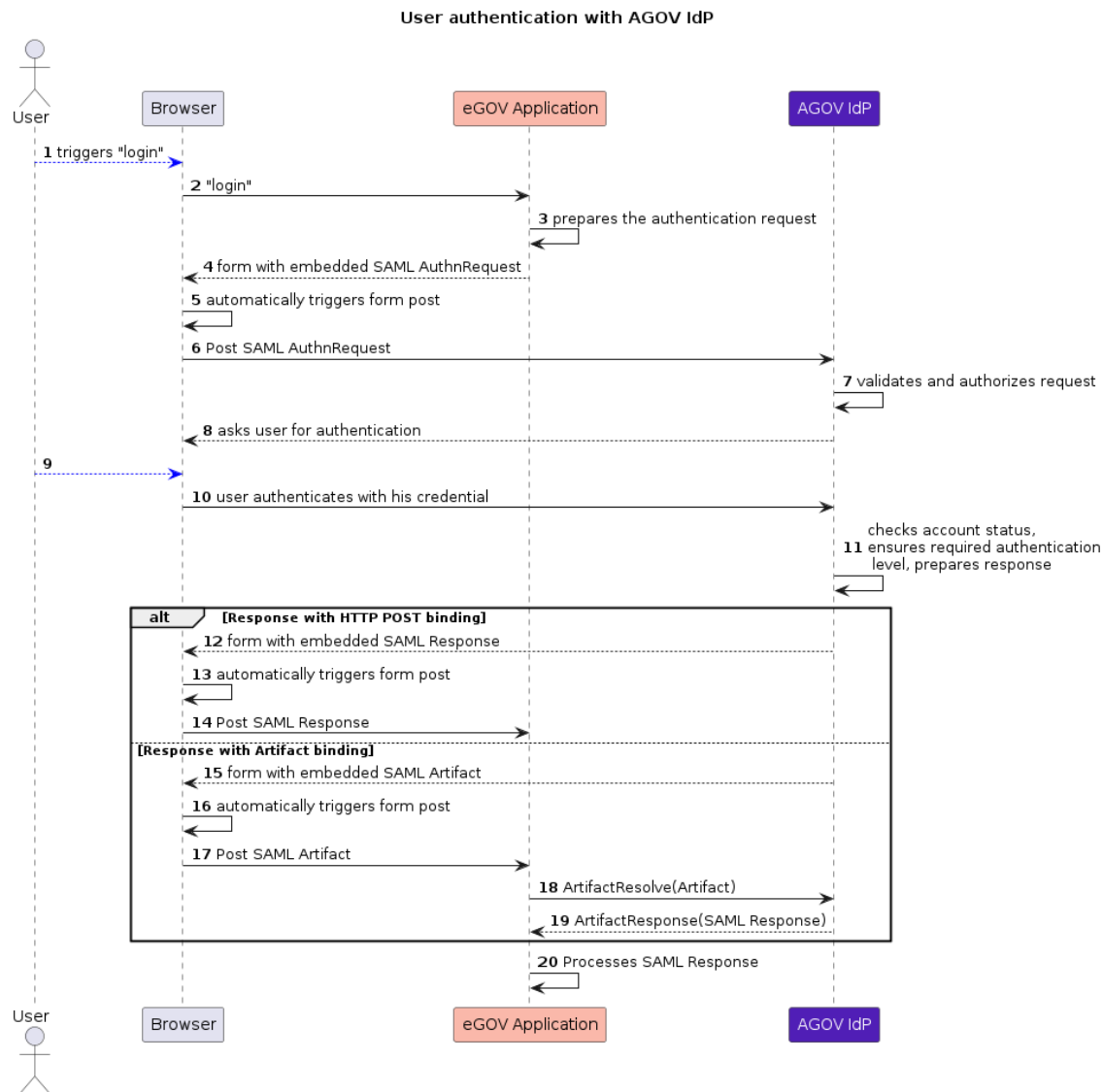
4.3 Runtime

4.3.1 Overview and supported bindings

AGOV IdP supports the following bindings:

- SAML HTTP-POST Binding (AuthnRequest, Response; urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST)
- SAML Artifact Binding (Response; urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Artifact;)

The following diagram shows how an eGov Application **MUST** proceed to authenticate the user using the SAML protocol.



4.3.2 SAML Entity ID and Endpoints

Endpoint	1.0
Entity ID	<code>https://idp.agov.admin.ch</code>
Metadata	<code>https://idp.agov.admin.ch/api/v1/metadata</code>
SingleSignOnService	<code>https://idp.agov.admin.ch/adfs/ls</code>
ArtifactResolutionService	<code>https://idp.agov.admin.ch/api/v1/saml/arp</code>

4.3.3 SAML AuthnRequest

eGov-Applications **MUST** submit an `<saml2p:AuthnRequest>` to the AGOV's idp SingleSignOnService endpoint which meets the following requirements:

Element / Attribute	Description	Availability
[ID]	The <code>ID</code> attribute in the root element MUST be present in each message. The value MUST be unique in the message. It is used as a reference in the signature of the message.	mandatory
[Destination]	The <code>destination</code> attribute in the root element of each message MUST always be present. Its value MUST be the URL to which the message is sent (i.e. the SSO endpoint URL of AGOV idp).	mandatory
[IssueInstant]	The <code>IssueInstant</code> attribute in the root element of each message MUST always be present. Its value indicates the time when the message was created. This MUST be encoded in UTC.	mandatory
[Version]	The <code>version</code> attribute in the root element of each message MUST always be present. Its value MUST be 2.0.	mandatory

Element / Attribute	Description	Availability
<Issuer>	The value of the <saml2:Issuer> element MUST match the EntityID attribute of the eGov-Application that created the <AuthnRequest> and that was previously registered as SAML ServiceProvider with AGOV IdP.	mandatory
<saml2p:NameIDPolicy>	<saml2p:NameIDPolicy> SHOULD NOT be present. AGOV IdP emits an Assertion with a persistent NameID.	recommended
<saml2p:RequestedAuthnContext>/<AuthnContextClassRef>	<p>The authentication request MAY contain one <saml2p:RequestedAuthnContext> element with at most one <saml2:AuthnContextClassRef> element, if an authentication with a different minimal level is requested than defined during the registration of the eGov Application as SAML ServiceProvider with AGOV.</p> <p>The value of the <saml2:AuthnContextClassRef> element MUST be one of the following:</p> <ul style="list-style-type: none"> urn:qa.agov.ch:names:tc:ac:classes:100 urn:qa.agov.ch:names:tc:ac:classes:200 urn:qa.agov.ch:names:tc:ac:classes:300 urn:qa.agov.ch:names:tc:ac:classes:400 	optional
<saml2p:RequestedAuthnContext>/<AuthnContextClassRef>[Comparison]	The Attribute Comparison MUST be set to minimum IF the the element <saml2p:RequestedAuthnContext> is used.	conditional
[ForceAuthn] and [IsPassive]	The attributes ForceAuthn and IsPassive MUST NOT be used. The AGOV IdP always freshly authenticates the user (behavior as if ForceAuthn was set to true and IsPassive to false).	mandatory
[AssertionConsumerServiceURL]	<p>Specifies by value the location to which the <Response> message MUST be returned to the requester and MUST be set.</p> <p>The AssertionConsumerServiceURL MUST match an AssertionConsumerService element of the eGov Application previously registered as SAML ServiceProvider with AGOV,</p>	mandatory
[ProtocolBinding]	The value of the ProtocolBinding attribute MUST be urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST.	mandatory

Element / Attribute	Description	Availability
<ds:Signature>	The <AuthnRequest> MUST be digitally signed with a certificate of the eGov Application previously registered as SAML ServiceProvider with AGOV (<ds:Signature> element included).	mandatory

All other optional elements of an `AuthnRequest` as specified in [\[SAML-CORE\]](#) **SHOULD NOT** be present. The AGOV IdP will ignore them.

The eGov Application **MAY** also transmit a RelayState together with the AuthnRequest to the AGOV IdP. If used, the RelayState **MUST NOT** exceed 80 bytes.

4.3.4 SAML Response

4.3.4.1 Content

The AGOV IdP **MUST** answer the AuthnRequest with a SAML Response respecting the following points:

Element / Attribute	Description	Availability
[ID]	The <code>ID</code> attribute in the root element MUST be present in each message. The value MUST be unique in the message. It is used as a reference in the signature of the message.	mandatory
[Destination]	The <code>destination</code> attribute in the root element of each message MUST always be present. Its value MUST be the URL to which the message is sent (i.e. the <code>AssertionConsumerServiceURL</code> specified in the <code>AuthnRequest</code> by the eGov Application).	mandatory
[IssueInstant]	The <code>IssueInstant</code> attribute in the root element of each message MUST always be present. Its value indicates the time when the message was created. This MUST be encoded in UTC.	mandatory
[Version]	The <code>version</code> attribute in the root element of each message MUST always be present. Its value MUST be 2.0.	mandatory
<Issuer>	The value of the <saml2:Issuer> element MUST match the <code>EntityID</code> attribute of the AGOV IdP.	mandatory
[InResponseTo]	The <code>ID</code> of a SAML protocol message in response to which an attesting entity can present the assertion. The values of the ID attribute in a request and the InResponseTo attribute in the corresponding response MUST match.	mandatory

Element / Attribute	Description	Availability
<Status>	<p>A code representing the <code>status</code> of the corresponding request.</p> <p>The <code><saml2p:Response></code> element MUST contain a <code><saml2p:Status></code> element. That element MUST contain a <code><saml2p:StatusCode></code>.</p> <p>If the authentication request was not successful, the <code><saml2p:Status></code> element MUST contain an error message and there will be no assertion.</p> <p>The value of the <code><saml2p:StatusMessage></code> element MUST contain an error ID, which can be traced back on demand (i.e. "<code><some error describing text> Request ID: <ID></code>").</p>	mandatory
<saml2:Assertion> or <saml2:EncryptedAssertion>	<p>In case of a successful authentication request (top-level <code>StatusCode = urn:oasis:names:tc:SAML:2.0:status:Success</code>), the <code><saml2p:Response></code> element MUST contain a <code><saml2:Assertion></code> element or optionally an encrypted assertion.</p> <p>If the eGov Application registered a public key for encryption with AGOV IdP, then the assertion will be encrypted.</p>	conditional
<ds:Signature>	<p>The <code><saml2p:Response></code> MUST be digitally signed with a certificate of the AGOV IdP published in its metadata (<code><ds:Signature></code> element included).</p>	mandatory

4.3.4.2 Error cases

There are cases, in which the AGOV IdP will not return a SAML Response to the requester:

- If the AuthnRequest is issued by a **not** previously registered eGov Application, if the signature is invalid, or if the AssertionConsumerServiceURL is missing or doesn't match one of the URLs registered for the eGov application, then no Response will be returned and an error page will be displayed to the user in the browser.
- If the user lets time out the authentication. This might happen because the user is interrupted and leaves the page, closes the browser, or because he didn't have an AGOV account or not one meeting the requested level of assurance, and tries to register one or upgrade it.

Reasons for not successful authentications can be:

- The IdP cannot fulfill the valid `AuthnRequest` (top-level `StatusCode = urn:oasis:names:tc:SAML:2.0:status:Responder`) because of
 - The user has canceled the authentication (second-level `StatusCode = urn:oasis:names:tc:SAML:2.0:status:AuthnFailed`)
 - The user was successfully authenticated, but the requested quality level could not be met and the user denied to upgrade his account (second-level `StatusCode = urn:oasis:names:tc:SAML:2.0:status:NoAuthnContext`)

- If the AuthnRequest is issued by an authorized eGov application, but contains invalid elements (top-level `StatusCode = urn:oasis:names:tc:SAML:2.0:status:Requester`).

4.3.4.3 HTTP-POST Binding

The AGOV IdP **MUST** return the SAML Response through the POST Binding to the eGov Application which sent the AuthnRequest.

The AGOV IdP **MUST** return the RelayState together with the SAML Response as received with the AuthnRequest.

4.3.4.4 HTTP-Artifact Binding

The AGOV IdP **MUST** answer the AuthnRequest with a SAML Artifact together with the RelayState as received with the AuthnRequest.

The eGov Application **MUST** validate that the SourceID contained in the Artifact belongs to AGOV IdP and **MUST** then call the artifact resolution endpoint with an ArtifactResolve message containing the following elements:

Element / Attribute	Description	Availability
[ID]	The <code>ID</code> attribute in the root element MUST be present in each message. The value MUST be unique in the message. It is used as a reference in the signature of the message.	mandatory
[Destination]	The <code>destination</code> attribute in the root element of each message MUST always be present. Its value MUST be the URL to which the message is sent (i.e. the SSO endpoint URL of AGOV idp).	mandatory
[IssueInstant]	The <code>IssueInstant</code> attribute in the root element of each message MUST always be present. Its value indicates the time when the message was created. This MUST be encoded in UTC.	mandatory
[Version]	The <code>version</code> attribute in the root element of each message MUST always be present. Its value MUST be 2.0.	mandatory
<Issuer>	The value of the <code><saml2:Issuer></code> element MUST match the <code>EntityID</code> attribute of the eGov-Application that created the <code><AuthnRequest></code> and that was previously registered as SAML ServiceProvider with AGOV IdP.	mandatory
<Artifact>	The SAML Artifact received from the IdP	mandatory
<ds:Signature>	The <code><saml2p:ArtifactResolve></code> MUST be digitally signed with a certificate of the eGov Application previously registered as SAML ServiceProvider with AGOV (<code><ds:Signature></code> element included).	mandatory

The AGOV IdP **MUST** answer the ArtifactResolve request with an ArtifactResponse respecting the following points:

Element / Attribute	Description	Availability
[ID]	The <code>ID</code> attribute in the root element MUST be present in each message. The value MUST be unique in the message. It is used as a reference in the signature of the message.	mandatory
[IssueInstant]	The <code>IssueInstant</code> attribute in the root element of each message MUST always be present. Its value indicates the time when the message was created. This MUST be encoded in UTC.	mandatory
[Version]	The <code>version</code> attribute in the root element of each message MUST always be present. Its value MUST be 2.0.	mandatory
<Issuer>	The value of the <code><saml2:Issuer></code> element MUST match the <code>EntityID</code> attribute of the AGOV IdP.	mandatory
[InResponseTo]	The <code>ID</code> of a SAML protocol message in response to which an attesting entity can present the assertion. The values of the <code>ID</code> attribute in a request and the <code>InResponseTo</code> attribute in the corresponding response MUST match.	mandatory
<Status>	A code representing the <code>status</code> of the corresponding request. The <code><saml2p:ArtifactResponse></code> element MUST contain a <code><saml2p:Status></code> element. That element MUST contain a <code><saml2p:StatusCode></code> .	mandatory
<saml2:Response>	In case of a successful artifact request (top-level <code>StatusCode = urn:oasis:names:tc:SAML:2.0:status:Success</code>), the <code><saml2p:Response></code> element MUST contain a <code><saml2:Response></code> element matching the artifact ID. If the eGov Application registered a public key for encryption with AGOV IdP, then the assertion will be encrypted.	conditional

There are cases, in which the AGOV IdP will not return a SAML Response to the requester:

- If the Artifact is unknown by the IdP
- If the AuthnRequest resulting in the SAML Response wasn't issued by the eGov Application which sent the ArtifactResolve request.

4.3.5 SAML Assertion


The AGOV IdP will emit a SAML Assertion respecting the following points:

Element / Attribute	Description	Availability
[ID]	The <code><saml2:Assertion></code> element MUST contain an <code>ID</code> attribute.	mandatory
[IssueInstant]	The <code><saml2:Assertion></code> element MUST contain an <code>IssueInstant</code> attribute. The <code>IssueInstant</code> value MUST be set to the time when the assertion was issued.	mandatory
[Version]	The <code>version</code> attribute in the root element of each message MUST always be present. Its value MUST be 2.0.	mandatory
<Issuer>	The value of the <code><saml2:Issuer></code> element MUST match the <code>EntityID</code> attribute of the AGOV IdP.	mandatory
<ds:Signature>	The <code><saml2:Assertion></code> MUST be digitally signed with a certificate of the AGOV IdP published in its metadata (<code><ds:Signature></code> element included).	mandatory
<Subject>	The <code><saml2:Assertion></code> element MUST contain a <code><saml2:Subject></code> element.	mandatory
<Subject>/NameID	The <code><saml2:Subject></code> element MUST contain a <code><saml2:NameID></code> element. Its value is the AGOV account ID. This is a persistent identifier, which remains unchanged over the whole lifecycle of the account. The <code><saml2:Subject></code> element MUST contain an attribute <code>Format</code> . Its value MUST be <code>'urn:oasis:names:tc:SAML:2.0:nameid-format:persistent'</code>	mandatory
<Subject>/<SubjectConfirmation>	The <code><saml2:Subject></code> element MUST contain a <code><saml2:SubjectConfirmation></code> element. The <code><saml2:SubjectConfirmation></code> element MUST contain an attribute <code>"Method"</code> . Its value MUST be <code>urn:oasis:names:tc:SAML:2.0:cm:bearer</code> . The <code><saml2:SubjectConfirmation></code> element MUST contain a <code><saml2:SubjectConfirmationData></code> element. That element MUST have attributes <code>InResponseTo</code> , <code>Recipient</code> and <code>NotOnOrAfter</code> . The values of the <code>ID</code> attribute in a request and the <code>InResponseTo</code> attribute in the corresponding assertion MUST match. The value of the <code>NotOnOrAfter</code> attribute MUST be set to the value of the <code>IssueInstant</code> plus 30 seconds. This allows a timeframe of 30 seconds for the message transmission and the validation of the <code>SubjectConfirmation</code> .	mandatory

Element / Attribute	Description	Availability
	The <code>Recipient</code> attribute MUST contain the assertion consumer URL of the eGov Application - the URL to which the assertion had been sent.	
<Conditions>	<p>The <code><saml2:Assertion></code> element MUST contain a <code><saml2:Conditions></code> element.</p> <p>It MUST have a <code>NotBefore</code> und <code>NotOnOrAfter</code> attribute.</p> <p>The value of the <code>NotBefore</code> attribute MUST be set to the value of the <code>IssueInstant</code>.</p> <p>The value of the <code>NotOnOrAfter</code> MUST be set to the value of the <code>IssueInstant</code> plus 4 hours. This doesn't mean that all statements contained in the assertion will</p> <ul style="list-style-type: none"> • remain correct and accurate throughout this period • become incorrect or inaccurate after this period. <p>Furthermore, it MUST contain a <code><saml2:AudienceRestriction></code> element.</p>	mandatory
<Conditions>/ <AudienceRestriction>	<p>The <code><saml2:AudienceRestriction></code> element MUST contain a <code><saml:Audience></code> element.</p> <p>The <code><saml:Audience></code> element MUST match the EntityID attribute of the eGov-Application that created the <code><AuthnRequest></code> and that was previously registered as SAML ServiceProvider with AGOV IdP.</p>	mandatory
<AuthnStatement>	<p>The <code><saml2:Assertion></code> element MUST contain exactly one <code><saml2:AuthnStatement></code> element.</p> <p>This element MUST contain an <code>AuthnInstant</code>, a <code>SessionIndex</code> and a <code><saml2:AuthnContext></code> element.</p> <p>The <code>AuthnInstant</code> MUST be set to the time where the user was authenticated.</p> <p>The <code>SessionIndex</code> MUST be present, as some SAML relying parties (aka Service Providers or eGov application) have problems, if he is missing. However the IdP will not maintain a global session.</p>	mandatory
<AuthnStatement>/ <AuthnContext>	<p>The <code><saml2:AuthnContext></code> element MUST contain a <code><saml2:AuthnContextClassRef></code> element.</p> <p>If the authentication was not performed by the AGOV IdP himself, but delegated to a federated eGov IdP, then the <code><saml2:AuthnContext></code> element MUST contain also an <code><saml2:AuthenticatingAuthority></code> element. It's value is the EntityID of the IdP, which actually performed the authentication.</p>	mandatory conditional

Element / Attribute	Description	Availability
<AttributeStatement>	The <saml2:Assertion> element MUST contain a <saml2:AttributeStatement> element.	mandatory
<AttributeStatement>/ <Attribute>	The <saml2:AttributeStatement> element MUST contain one or more <saml2:Attribute> elements. Each element can contain one or more <saml2:AttributeValue> elements.	mandatory

The SAML Assertion will contain the following attributes:

Name	Description	Availability
http://schemas.agov.ch/ws/2025/07/identity/claims/op/conversationId	technical unique identifier to be able to trace issuance of the assertion in the AGOV infrastructure	mandatory
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress	the main email address linked to the AGOV account of the user. This email is used as account identifier by the user.  the user may change the email over time.	mandatory
http://schemas.agov.ch/ws/2023/05/identity/claims/languageOfCorrespondance	language of correspondence as defined by the user. ISO 639-1 two letter code in lower case. Possible value: de, fr, it, en, rm (eCH-0011:languageType)	mandatory
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/givenname	given name(s) of the account holder	mandatory
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/surname	official name(s) of the account holder	mandatory
http://schemas.agov.ch/ws/2023/05/identity/claims/dateOfBirth	date of birth of the account holder xs:date ("[-]CCYY-MM-DD[Z (+ -)hh:mm]" following ISO 8601 with an optional time zone)	mandatory
http://schemas.agov.ch/ws/2023/05/identity/claims/sex	sex of the account holder 1: male, 2: female oder 3: undetermined (eCH-0044:sexType)	mandatory

Name	Description	Availability
http://schemas.agov.ch/ws/2023/05/identity/claims/nationality	country of nationality two letter alphanumeric code as of ISO 3166.	mandatory
http://schemas.agov.ch/ws/2023/05/identity/claims/socialSecurityNumber	social security insurance number of the user. 13 digits number (representation without dots, eCH-0044:vnType). <i>conditional</i> : only available for AGOV accounts with an AGOVaq 400 and if the application was entitled to receive it during registration with AGOV connect.	optional
http://schemas.agov.ch/ws/2023/05/identity/claims/placeOfBirth	place of birth of the user.	optional
http://schemas.agov.ch/ws/2023/05/identity/claims/eldNumber	the unique identifier of the Swiss E-ID used by the user. This value is only available for users which are using their E-ID with AGOV IdP.	optional
http://schemas.agov.ch/ws/2023/05/identity/claims/qa/dateOfVerification	date of identity verification, which lead to or confirmed the level stated in the <code><saml2:AuthnContextClassRef></code> element. <code>xs:dateTime (»[-]CCYY-MM-DDThh:mm:ss[Z (+ -)hh:mm]«</code> following ISO 8601 with an optional time zone) <i>conditional</i> : only available for level 200 and above (<code>>=urn:qa.agov.ch:names:tc:ac:classes:200</code>)	conditional
http://schemas.agov.ch/ws/2023/05/identity/claims/qa/validTillDate	date when the identity verification, which lead to or confirmed the level stated in the <code><saml2:AuthnContextClassRef></code> element, will expire and the account holder will have to undergo a new verification to keep the level. <code>xs:dateTime (»[-]CCYY-MM-DDThh:mm:ss[Z (+ -)hh:mm]«</code> following ISO 8601 with an optional time zone)	conditional

Name	Description	Availability
	<i>conditional</i> : only available for level 200 and above (<code>>=urn:qa.agov.ch:names:tc:ac:classes:200</code>)	
http://schemas.agov.ch/ws/2023/05/identity/claims/qa/verificationMethod	method used, to verify the users identity. possible values: <code>None</code> , <code>SimpleLetter</code> , <code>Video</code> , <code>Bmid</code> , <code>Counter</code> , <code>Eid</code> , <code>OtherIdP</code>	mandatory
http://schemas.agov.ch/ws/2023/08/identity/claims/address/street http://schemas.agov.ch/ws/2023/08/identity/claims/address/houseNumber http://schemas.agov.ch/ws/2023/08/identity/claims/address/zipCode http://schemas.agov.ch/ws/2023/08/identity/claims/address/town http://schemas.agov.ch/ws/2024/02/identity/claims/address/country http://schemas.agov.ch/ws/2024/02/identity/claims/address/countryName http://schemas.agov.ch/ws/2023/05/identity/claims/address/qa/verificationMethod	the address of the user. Empty attributes are missing in the assertion. Note: <ul style="list-style-type: none"> country: two letter alphanumeric code as of ISO 3166. countryName: the name of the country in the language of correspondence of the user verificationMethod: <code>None</code>, <code>LocationCertified</code>, <code>DomicileCertified</code>, <code>SimpleLetter</code>, <code>Bmid</code> <i>conditional</i> : only available for AGOV accounts with an AGOVaq 200 or higher and if the application was entitled to receive it during registration with AGOV connect.	conditional

4.3.6 SAML Response and Assertion processing

AGOV highly recommends that eGov Applications follows the recommendations of the OWASP SAML cheat sheet [\[OWASP-SAML\]](#).

5 OIDC Interface

5.1 References

5.1.1 Specifications

[OIDC-CORE]	OpenID Connect CORE Version 1.0 incorporating errata set 1, November 8, 2014 https://openid.net/specs/openid-connect-core-1_0.html
[OIDC-DSCVR]	OpenID Connect Discovery Version 1.0 incorporating errata set 1, November 8, 2014 https://openid.net/specs/openid-connect-discovery-1_0.html
[OIDC-LOGOUT]	OpenID Connect RP-Initiated Logout Version 1.0, September 12, 2022 https://openid.net/specs/openid-connect-rpinitiated-1_0.html
[OAUTH]	The OAuth 2.0 Authorization Framework October 2012 https://datatracker.ietf.org/doc/html/rfc6749
[OAUTH-NATIV]	OAuth 2.0 for Native Apps October 2017 https://datatracker.ietf.org/doc/html/rfc8252
[OAUTH-SBCP]	OAuth 2.0 Security Best Current Practice Draft version from 2023-03-13 https://datatracker.ietf.org/doc/draft-ietf-oauth-security-topics/22/
[JWT]	JSON Web Token (JWT) May 2015 https://datatracker.ietf.org/doc/html/rfc7519

[JWS]	JSON Web Signature (JWS) May 2015 https://datatracker.ietf.org/doc/html/rfc7515
[JWA]	JSON Web Algorithms (JWA) May 2015 https://datatracker.ietf.org/doc/html/rfc7518
[PKCE]	Proof Key for Code Exchange by OAuth Public Clients rfc7636, September 2015 https://datatracker.ietf.org/doc/rfc7636/

5.2 Defintime

eGov-Applications which want to use AGOV for authentication with OIDC **MUST** register with AGOV IdP before using it. Registration will be done through the selfservice portal AGOV connect.

Note: AGOV connect doesn't support dynamic registration.

The following application parameters **MUST** be provided during registration:

Parameter	Description	Availability
application_type	Either <code>native</code> or <code>web</code>	optional
client_type	Either <code>public</code> or <code>private</code> .	mandatory
redirect_uris	Array of redirection URIs used by the application. Used by the IdP to validate the authentication request. If the <code>application_type</code> is <code>web</code> , then the URI scheme must be <code>https</code> .	mandatory
client_name	Name to be displayed to the user on the login screen.	optional
logo_uri	Logo to be added on the login screen.	optional
token_endpoint_auth_method	One out of <code>client_secret_post</code> , <code>client_secret_basic</code> , or <code>private_key_jwt</code> for private clients.	conditional

Parameter	Description	Availability
public key	if the method is <code>private_key_jwt</code> .	conditional
default_acr_values	The minimal level of assurance required for that application. Might be overwritten in the authentication request.	mandatory
scope	The special scopes address and svnr must be pre-declared during registration (see OIDC-Scopes below): <ul style="list-style-type: none"> svnr: the client would like to receive the user's social security number address: the client would like to receive the user's domicile address 	optional

The following parameters **MUST** be made available by AGOV connect to the Technical AppOwner:

Parameter	Description	Availability
client_id	Unique Client Identifier to be used in the authentication requests.	mandatory
client_secret	If the the client is <code>private</code> and the method not <code>private_key_jwt</code> then AGOV connect will generate a secret to be used in the token requests.	conditional

5.2.1 Support of public clients

AGOV IdP **MUST** support public clients.

The ID-Token is more exposed on public clients, than on private ones. Thus AGOV IdP **MUST** impose the following **restrictions** on public clients:

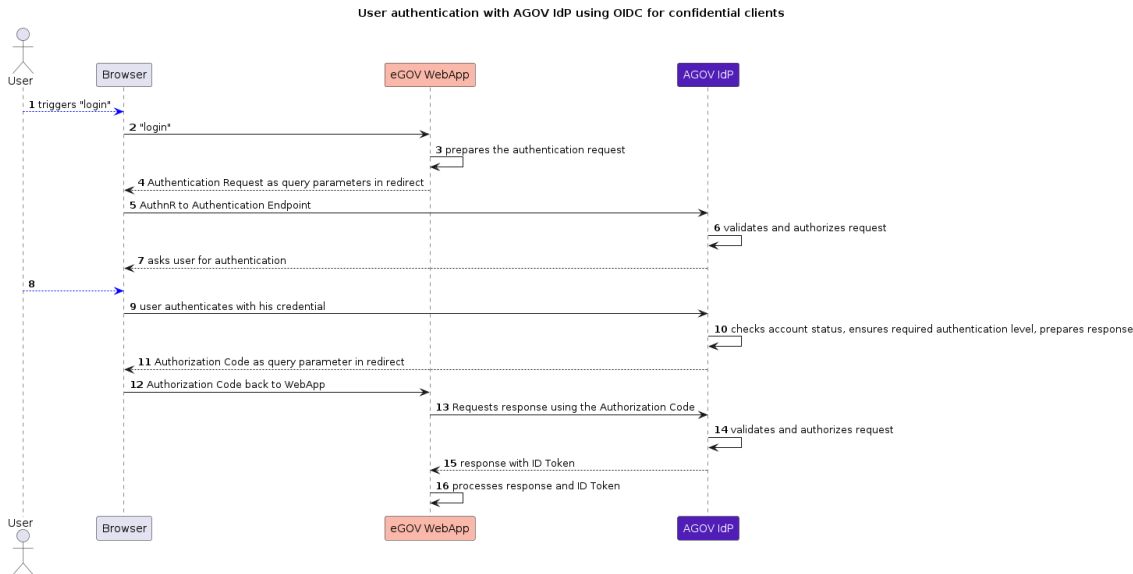
- Only a subset of Scopes is available (see [OIDC-Scopes](#) below)
- Public clients can't enforce a higher level of assurance for the accounts

5.3 Runtime

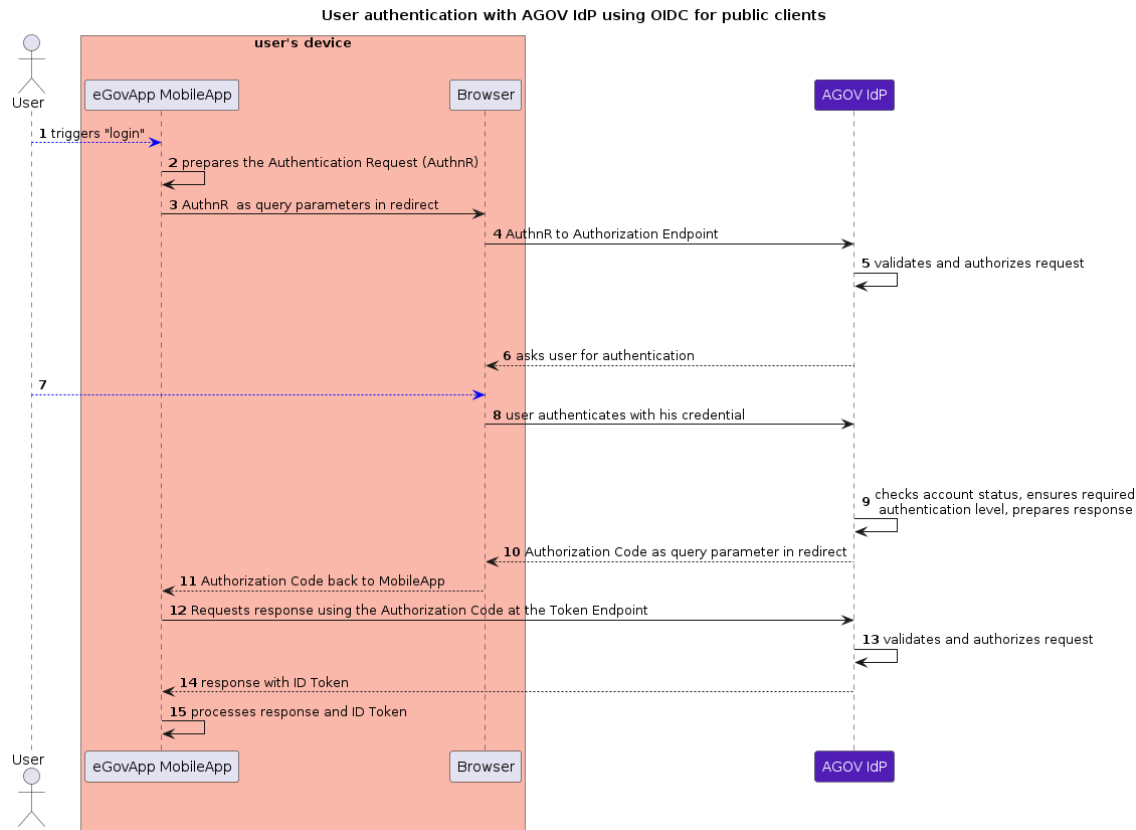
5.3.1 Supported Flows

AGOV IdP (in the OIDC context also called OpenID Provider) supports OIDC with the Authorization Code Flow.

The following sequence diagram shows the flow, as used by an eGov WebApp. This is an application where the authentication request and response is processed on the server side (in terms of Oauth a client of type "confidential web application") and the user interface of the application is in a web browser.



The following sequence diagram shows the flow, as used by an eGov MobileApp. This is an application which is installed and executed on the user's device (in terms of Oauth a client of type "public native application or public browser-based application").



5.3.2 Usage of UserInfo endpoint

The eGov applications should simply use the ID Token, which contains all relevant user attributes as requested in the authentication request. The ID Token is signed, and the signature can be verified against the public keys published as JSON Web Key Sets (URI available over the AGOV IdP's OpenID provider configuration, see below).

Some IAM products don't use the ID Token with the default settings (i.e. Keycloak). They use the Access Token with the UserInfo-endpoint to fetch the user's attribute. In order to ease the integration, AGOV allows this pattern. However, the validity of the Access Token is reduced to 60 seconds, which is just enough to allow one call to the UserInfo endpoint.

It is strongly recommended to work directly with the ID Token instead.

5.3.3 eGov Application initiated Logout

As mentioned above AGOV IDP keeps the session for one minute. If an eGov Application using OIDC wants to make sure the user is reauthenticated during that period, it can call the end session endpoint before.

For that the user agent must be redirected to the following URL:

```
https://idp.agov.admin.ch/logout?id_token_hint=ID_TOKEN_BASE64_VALUE&post_logout_redirect_uri=LOGOUT_URL
```

The following mandatory attributes must be passed as query parameters:

- `ID_TOKEN_BASE64_VALUE`: The users ID Token base64 encoded
- `LOGOUT_URL`: The url the AGOV IDP should redirect after the logout. This url has to be configured in AGOV IDP as well (AGOV Connect OIDC redirect uris).

5.3.4 Endpoints

The following endpoints are available:

Endpoint	PROD
Issuer Identifier	<code>https://idp.agov.admin.ch/</code>
OpenID Provider Configuration	
Authorization	<code>https://idp.agov.admin.ch/oauth2/authorize</code>
Token	<code>https://idp.agov.admin.ch/oauth2/token</code>
UserInfo	<code>https://idp.agov.admin.ch/userinfo</code>
End Session	<code>https://idp.agov.admin.ch/logout</code>

5.3.5 Authentication Request

The eGov OIDC Client must submit an Authentication Request to the Authorization Endpoint of AGOV IdP. The client can use either an http GET or POST method.

The request must meet the following requirements:

Parameter	Description	Availability
scope	The <code>scope</code> parameter MUST be present and contain the value <code>openid</code> . Additional scopes MAY be requested. See table below for the supported scopes.	mandatory
response_type	The <code>response_type</code> parameter MUST be set to the value <code>code</code> .	mandatory
client_id	The <code>client_id</code> parameter MUST be set to the value obtained during the registration of the client with AGOV IdP.	mandatory
redirect_uri	The <code>redirect_uri</code> parameter MUST be set and exactly match on of the URI's registred during registration of the client with AGOV IdP.	mandatory
state	The <code>state</code> parameter MAY be used by the client. From the AGOV IdP point of view, it's an opaque value, which if set, will be returned as is in the Response.	recommended
code_challenge	The <code>code_challenge</code> parameter MUST be set according [PKCE] . The challenge MUST be prepared according the S256 method (see also below). Note: usage of PKCE is recommended for all type of clients, but mandatory for public clients.	recommended/mandatory
code_challenge_method	The <code>code_challenge</code> parameter MUST be set to the value <code>S256</code> .	mandatory
nonce	The <code>nonce</code> parameter MAY be used by the client. If a value is set, the AGOV IdP MUST add it unmodified to the issued ID Token.	recommended

Parameter	Description	Availability
acr_values	<p>The Authentication Request MAY contain (at most) one Authentication Context Class Reference value, if an authentication with a different minimal level is requested than defined during the registration of the eGov Application as OIDC client with AGOV.</p> <p>The value of the acr_values element MUST be one of the following:</p> <ul style="list-style-type: none"> • urn:qa.agov.ch:names:tc:ac:classes:100 • urn:qa.agov.ch:names:tc:ac:classes:200 • urn:qa.agov.ch:names:tc:ac:classes:300 • urn:qa.agov.ch:names:tc:ac:classes:400 	optional

All other optional parameters of an Authentication Request as described in [\[OIDC-CORE\]](#) **SHOULD NOT** be used. The AGOV IdP may ignore them or refuse to process the request.

Especially the `claims` parameter cannot be used to request specific claims in the requests. Scopes shall be used for that purpose.

AGOV IdP **MUST** support the following scopes:

Scope	Description
openid	technical scope to identify the OIDC protocol
profile	This scope value requests access to the user's default profile claims
email	This scope value requests access to the email and email_verified claims
agovProfile	This scope value requests access to the user's additional AGOV specific profile claims
svnr	Special scope to request the user's social security number
address	Special scope to request the user's home address

Other scope values are not supported.

5.3.6 AGOV IdP Authenticates the User

The AGOV IdP **MUST** authenticate the User for each authentication request. There is no support for a "none prompt" mode.

Authentication will require user interaction. Thus the Authentication Request **MUST** be sent by the client through a web-browser to the Authorization Endpoint of the IdP. The browser **MUST** support Javascript and support the following of redirects as the AGOV IdP **MAY** delegate the authentication to other federated IdPs.

The AGOV IdP **MUST NOT** ask for the user consent to deliver an ID Token to the clients. The user gives an implicit consent by creating an AGOV account and using AGOV IdP to authenticate for a client.

In case of a successful authentication, the AGOV IdP returns an Authentication Response by a redirect to the `redirect_uri` from the request, by adding the following parameters:

Parameter	Description	Availability
<code>code</code>	The <code>code</code> parameter MUST be present.	mandatory
<code>state</code>	If the <code>state</code> parameter was present in the request, the AGOV IdP MUST return it unmodified in the Response.	conditional

There are cases, for which the AGOV IdP will not return a response to the client:

- `client_id` doesn't belong to a previously registered client
- `redirect_uri` doesn't match one of the previously registered `redirect_uris` for that client

In such cases, an error page **MUST** be displayed to the user.

Otherwise, if the authentication fails, or couldn't be processed, an Error Response with the following parameters **MUST** be returned to the client:

Parameter	Description	Availability
<code>error</code>	<p>The <code>error</code> parameter MUST be present.</p> <p>The AGOV IdP MUST use one of the following codes:</p> <ul style="list-style-type: none"> • <code>invalid_request</code> • <code>access_denied</code> 	mandatory

Parameter	Description	Availability
	<ul style="list-style-type: none"> • <code>unauthorized_client</code> • <code>unsupported_response_type</code> • <code>invalid_scope</code> • <code>server_error</code> • <code>temporarily_unavailable</code> • <code>request_not_supported</code> • <code>request_uri_not_supported</code> • <code>registration_not_supported</code> 	
<code>error_description</code>	The AGOV IdP MUST return a human-readable text description of the error. It MUST contain an error ID, which can be traced back on demand (i.e. "<some error describing text> Request ID: <ID>").	optional
<code>state</code>	If the <code>state</code> parameter was present in the request, the AGOV IdP MUST return it unmodified in the Response.	conditional

5.3.7 Token Request

The eGov OIDC Client **MUST** use the received code to exchange it in an ID Token by sending an appropriate Token Request to the AGOV IdP's Token Endpoint. The Token Request **MUST** contain the following parameters:

Otherwise, if the authentication fails, or couldn't be processed, an Error Response with the following parameters **MUST** be returned to the client:

Parameter	Description	Availability
<code>grant_type</code>	The <code>grant_type</code> parameter MUST be present with the value <code>authorization_code</code> .	mandatory
<code>code</code>	The <code>code</code> parameter MUST be present with the value received in the Authentication Response.	mandatory
<code>client_id</code>	The <code>client_id</code> parameter MUST be set to the value obtained during the registration of the client with AGOV IdP.	mandatory
<code>redirect_uri</code>	The <code>redirect_uri</code> parameter MUST be set to the exactly same value as in the authentication request.	mandatory

Parameter	Description	Availability
code_verifier	The <code>code_verifier</code> parameter MUST be set according [PKCE] . It MUST match the challenge used for the authentication request.	mandatory

Other parameters except those for the client authentication **MUST NOT** be used.

Confidential clients **MUST** authenticate the request with the registered secret during registration.

AGOV IdP **MUST** support the following methods: `client_secret_basic`, `private_key_jwt` (PREFERRED).

5.3.8 AGOV IdP processes the Token Request

AGOV IdP **MUST** authenticate and authorize the Token Request. If the request is valid, the AGOV IdP **MUST** return a Token Response to the client containing an Access Token (parameter `access_token`) and an ID Token (`id_token` parameter).


AGOV IdP **MUST** ensure that the `access_token` is opaque and can't be used to extract user attributes.

i Note currently the `access_token` has no practical use for the client, as it can't be used to access any resources within AGOV. This may change later on.


The ID Token contains the following claims, based on the requested scopes in the Authentication Request:

Claim	Description	Scope for clients		Availability
		private	public	
conversationId	technical unique identifier to be able to trace issuance of the token in the AGOV infrastructure.	openid	openid	mandatory
iss	The value of the <code>iss</code> claim MUST match the Issuer Identifier of the AGOV IdP.	openid	openid	mandatory
source_iss	If the authentication was not performed by the AGOV IdP himself, but delegated to a federated eGov IdP, then the <code>source_iss</code> claim MUST be present and contain the EntityID (<code>iss</code> claim) of the IdP, which actually performed the authentication.	agovProfile	n/a	conditional
sub	The <code>sub</code> claim MUST be present and contains the AGOV account ID of the authenticated user. This is a persistent identifier, which remains unchanged over the whole lifecycle of the account.	openid	openid	mandatory
aud	The <code>aud</code> claim MUST be present and contain the <code>client_id</code> of the client that submitted the authentication request.	openid	openid	mandatory

Claim	Description	Scope for clients		Availability
		private	public	
	Note: The <code>aud</code> claim is currently identical to the <code>azp</code> claim. Agov IdP might support in a future version that an eGov application can define additional identifiers for other audiences.			
<code>azp</code>	The <code>aud</code> claim MUST be present and contain the <code>client_id</code> of the client that submitted the authentication request.	openid	openid	mandatory
<code>iat</code>	The value of the <code>iat</code> claim MUST be present and set by the AGOV IdP to the time, when the ID Token is issued.	openid	openid	mandatory
<code>jti</code>	The value of the <code>jti</code> claim MUST be present and set by the AGOV IdP to a unique value (a case sensitive string). The "jti" claim can be used to prevent the JWT from being replayed.	openid	openid	mandatory
<code>exp</code>	The value of the <code>exp</code> claim MUST be present and set to the value of the <code>iat</code> claim plus 4 hours. This doesn't mean that all statements contained in the claim will <ul style="list-style-type: none"> • remain correct and accurate throughout this period • become incorrect or inaccurate after this period. But it can be seen as the recommended timespan after which an eGov-Application should authenticate the user again.	openid	openid	mandatory
<code>auth_time</code>	The value of the <code>auth_time</code> claim MUST be present and set by the AGOV IdP to the time, when the user was authenticated by the IdP.	openid	openid	mandatory
<code>acr</code>	The value of the <code>acr</code> claim MUST be present and set by the AGOV IdP to the level of assurance of the AGOV account at the time of authentication. If <code>acr_values</code> was present in the authentication request, then the AGOV IdP MUST ensure that the level is at least as high as requested. If no <code>acr_values</code> was present in the authentication request, then the AGOV IdP MUST ensure that the level is at least as high as defined for the client during registration with AGOV IdP.	openid	openid	mandatory

Claim	Description	Scope for clients		Availability
		private	public	
nonce	If nonce was present in the authentication request, then the AGOV IdP MUST ensure that the claim's value is the same as in the request.	openid	openid	conditional
email	The value of the <code>email</code> claim MUST be present and set by the AGOV IdP to the main email address linked to the AGOV account of the user. This email is used as account identifier by the user.  the user may change the email over time	email	email	mandatory for scope
email_verified	The value of the <code>email</code> claim MUST be present and set by the AGOV IdP to the value 'true'. Is always 'true' for AGOV, as AGOV systematically verifies the user's e-mail address.	email	n/a	mandatory for scope
given_name	The value of the <code>given_name</code> claim MUST be present and set by the AGOV IdP to the value of the given name(s) of the account holder.	profile	profile	mandatory for scope
family_name	The value of the <code>family_name</code> claim MUST be present and set by the AGOV IdP to the value of the official name(s) of the account holder.	profile	profile	mandatory for scope
birthdate	The value of the <code>birthdate</code> claim MUST be present and set by the AGOV IdP to the value of date of birth of the account holder. It is expressed as an ISO 8601:2004 YYYY-MM-DD format.	profile	n/a	mandatory for scope
gender	The value of the <code>gender</code> claim MUST be present and set by the AGOV IdP to the value of the sex of the account holder. The value MUST be one out of male, female or undetermined.	profile	n/a	mandatory for scope
language	The value of the <code>language</code> claim MUST be present and set by the AGOV IdP to the value of the language of correspondence as defined by the user for the account. ISO 639-1 two letter code in lower case. Possible value: <code>de</code> , <code>fr</code> , <code>it</code> , <code>en</code> , <code>rm</code> (eCH-0011:languageType)	profile	profile	mandatory for scope

Claim	Description	Scope for clients		Availability
		private	public	
locale	<p>The value of the <code>locale</code> claim MUST be present and set by the AGOV IdP to the value composed of the language and nationality of the account holder.</p> <p>It is represented as a BCP47 language tag (an ISO 639-1 Alpha-2 language code in lowercase and an ISO 3166-1 Alpha-2 country code in uppercase, separated by a dash).</p>	profile	profile	mandatory for scope
nationality	<p>The value of the <code>nationality</code> claim MUST be present and set by the AGOV IdP to the value of the country of nationality of the account holder.</p> <p>The value MUST be a two letter alphanumeric code as of ISO 3166.</p>	agovProfile	n/a	mandatory for scope
placeOfBirth	<p>The value of the <code>placeOfBirth</code> claim MAY be present and set by the AGOV IdP to the value of the place of birth of the user.</p> <p><i>conditional: only available for level 400 and above (>= urn:qa.agov.ch:names:tc:ac:classes:400)</i></p>	agovProfile	n/a	optional for scope
eIdNumber	<p>The value of the <code>eIdNumber</code> claim MAY be present and set by the AGOV IdP to the value of the unique identifier of the Swiss E-ID used by the user.</p> <p>This value is only available for users which are using their E-ID with AGOV IdP.</p> <p><i>conditional: only available for level 500 (urn:qa.agov.ch:names:tc:ac:classes:500)</i></p>	agovProfile	n/a	optional for scope
qa_DateOfVerification	<p>The value of the <code>qa_DateOfVerification</code> claim MAY be present and set by the AGOV IdP to the date of identity verification, which lead to or confirmed the level stated in the <code>acr</code> claim.</p> <p>It is a JSON number representing the number of seconds from 1970-01-01T0:0:0Z as measured in UTC</p> <p><i>conditional: only available for level 200 and above (>= urn:qa.agov.ch:names:tc:ac:classes:200)</i></p>	agovProfile	n/a	conditional for scope
qa_validTillDate	<p>The value of the <code>qa_validTillDate</code> claim MAY be present and set by the AGOV IdP to the value of date when the identity verification, which lead to or confirmed the level stated in the <code>acr</code> claim, will expire and the account holder will have to undergo a new verification to keep the level.</p>	agovProfile	n/a	conditional for scope

Claim	Description	Scope for clients		Availability
		private	public	
	<p>It is a JSON number representing the number of seconds from 1970-01-01T0:0:0Z as measured in UTC</p> <p><i>conditional</i>: only available for level 200 and above (\geq urn:qa.agov.ch:names:tc:ac:classes:200)</p>			
qa_verificationMethod	<p>The value of the <code>qa_verificationMethod</code> claim MUST be present and set by the AGOV IdP to the method used, to verify the user's identity.</p> <p>Possible values: <code>None</code>, <code>SimpleLetter</code>, <code>Video</code>, <code>Bmid</code>, <code>Counter</code>, <code>Eid</code>, <code>OtherIdP</code></p>	agovProfile	n/a	conditional for scope
socialSecurityNumber	<p>The value of the <code>socialSecurityNumber</code> claim MAY be present and set by the AGOV IdP to the value of social security insurance number of the user.</p> <p>It is a 13 digits number (representation without dots, eCH-0044:vnType).</p> <p><i>conditional</i>: only available for level 400 and above (\geq urn:qa.agov.ch:names:tc:ac:classes:400)</p>	svnr	n/a	conditional for scope
address	<p>The value of the <code>address</code> claim MAY be present and set by the AGOV IdP to the value of the address of the user.</p> <p>It is a json structure containing the address components <code>street_address</code>, <code>locality</code>, <code>postal_code</code>, <code>street</code>, <code>house_number</code>, <code>verified</code>, <code>verificationMethod</code>, <code>country</code>, <code>countryCode</code> and formatted.</p> <p><i>conditional</i>: only available for AGOV accounts with an AGOVaq of 200 or higher. To get the address the authentication request must request an AGOVaq of 200 or higher (either at registration of the client or with the <code>acr_values</code> parameter in the authorization request) and the client must be entitled to receive the address during registration.</p> <p> The address claim follows [OIDC-CORE] and is extended by</p> <ul style="list-style-type: none"> in addition to the <code>country</code> attribute, AGOV IdP returns the two letter alphanumeric code as of ISO 3166 as <code>countryCode</code>. 	address	n/a	conditional for scope

Claim	Description	Scope for clients		Availability
		private	public	
	<ul style="list-style-type: none"> • the attribute <code>verified</code> which is true, if and only if the address was verified • the attribute <code>verificationMethod</code> which contains the method used to verify the address, one of: <code>None</code>, <code>LocationCertified</code>, <code>DomicileCertified</code>, <code>SimpleLetter</code>, <code>Bmid</code>. Semantics: <ul style="list-style-type: none"> ○ <code>None</code>: the address isn't verified by any means ○ <code>LocationCertified</code>: the Swiss Post API returned a value of <code>CERTIFIED</code> meaning that the address is valid. However, there is no certification whatsoever about whether or not the person actually lives there. ○ <code>DomicileCertified</code>: the Swiss Post API returned a value of <code>DOMICILE_CERTIFIED</code> meaning that the Swiss Post knows that the person is living at that address. ○ <code>SimpleLetter</code>: AGOV successfully sent an A-Post letter to that person at that address. ○ <code>Bmid</code>: AGOV successfully sent a BmID (Brief mit Id Check) that that person at that address. • the two attributes <code>street</code> and <code>house_number</code>, which are the decomposed <code>street_address</code>. • <code>formatted</code> doesn't contain a title and user's names. 			

5.3.9 Response and ID Token processing

The signature of the received ID Token **MUST** be validated against the keys published in the JSON Web Key Set of the AGOV IdP. Its URI can be obtained by reading the `jwtks_uri` claim in the AGOV IdPs OpenID Provider Configuration.

If a nonce was presented to AGOV IdP in the Authentication Request, then ID token must contain a claim with the same value.

6 Appendix

6.1 Attribute lengths

The following maximal length of string attributes will not be exceeded in the issued tokens by the AGOV IdP:

Attribute		Length	Notes
SAML	OIDC		
Subject/Nameld	sub	36	AGOV account ID is a UUID
http://schemas.agov.ch/ws/2025/07/identity/claims/op/conversationId	conversationId	32	
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress	email	255	
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/givenname	given_name	50	
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/surname	family_name	100	
http://schemas.agov.ch/ws/2023/05/identity/claims/placeOfBirth	placeOfBirth	50	
http://schemas.agov.ch/ws/2023/08/identity/claims/address/street	address.street	60	
http://schemas.agov.ch/ws/2023/08/identity/claims/address/houseNumber	address.house_number	12	
http://schemas.agov.ch/ws/2023/08/identity/claims/address/zipCode	address.postal_code	10	
http://schemas.agov.ch/ws/2023/08/identity/claims/address/town	address.locality	50	
http://schemas.agov.ch/ws/2024/02/identity/claims/address/countryName	address.country	75	
n/a	address.street_address	73	
n/a	address.formatted	370	

6.2 Sample messages

6.2.1 SAML

The following block shows an AuthnRequest sent by the eGov application with the entityID "https://samlsp.egov.sample.gov/test/api/saml2/service-provider-metadata/agovidp" to the AGOV IdP:

```

<saml2p:AuthnRequest xmlns:saml2p="urn:oasis:names:tc:SAML:2.0:protocol"
  AssertionConsumerServiceURL="https://samlsp.egov.sample.gov/login/saml2/sso/agovidp"
  Destination="https://idp.agov.admin.ch/adfs/ls"
  ID="ARQff9160d-8db6-4838-86cf-2a41a5808b48"
  IssueInstant="2023-08-29T07:56:25.183Z"
  ProtocolBinding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
  Version="2.0">
  <saml2:Issuer xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion">https://samlsp.egov.sample.gov/saml2/service-provider-
  metadata/agovidp</saml2:Issuer>
  <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <ds:SignedInfo>
      <!-- removed -->
    </ds:SignedInfo>
    <ds:SignatureValue>
      <!-- removed -->
    </ds:SignatureValue>
    <ds:KeyInfo>
      <ds:X509Data>
        <ds:X509Certificate>
          <!-- removed -->
        </ds:X509Certificate>
      </ds:X509Data>
    </ds:KeyInfo>
  </ds:Signature>
  <saml2p:RequestedAuthnContext Comparison="minimum">
    <saml2:AuthnContextClassRef
  xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion">urn:qa.agov.ch:names:tc:ac:classes:300</saml2:AuthnContextClassRef>
    </saml2p:RequestedAuthnContext>
</saml2p:AuthnRequest>
  
```

i Note: the eGov application was registered with AGOV connect and declared to need the address and svnr.

The AGOV IdP validated the requested, authenticated the user with the AGOV account ID "6b113b9d-1376-4583-9628-3f9224d2c68e" and sent the following SAML response back to the eGov application:

```

<saml2p:Response xmlns:saml2p="urn:oasis:names:tc:SAML:2.0:protocol"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  Consent="urn:oasis:names:tc:SAML:2.0:consent:unspecified"
  Destination="https://samlsp.egov.sample.gov/login/saml2/sso/agovidp"
  ID="_ceb65c9e158abf5f44eeca71fea6239c"
  InResponseTo="ARQff9160d-8db6-4838-86cf-2a41a5808b48"
  IssueInstant="2023-08-29T07:56:40.330Z"
  Version="2.0">
  <saml2:Issuer xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion">https://idp.agov.admin.ch/</saml2:Issuer>
  <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <ds:SignedInfo>
      <!-- removed -->
    </ds:SignedInfo>
    <ds:SignatureValue>
      <!-- removed -->
    </ds:SignatureValue>
    <ds:KeyInfo>
      <ds:X509Data>
        <ds:X509Certificate>
          <!-- removed -->
        </ds:X509Certificate>
      </ds:X509Data>
    </ds:KeyInfo>
  </ds:Signature>
  <saml2p:Status>
    <saml2p:StatusCode Value="urn:oasis:names:tc:SAML:2.0:status:Success"/>
  </saml2p:Status>
  <saml2:Assertion xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion"
    xmlns:a="http://schemas.xmlsoap.org/ws/2009/09/identity/claims"
    ID="_4957f0791b64f465e69699d551c747e7"
    IssueInstant="2023-08-29T07:56:40.331Z"
    Version="2.0">
    <saml2:Issuer>https://idp.agov.admin.ch/</saml2:Issuer>
    <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
      <ds:SignedInfo>
        <!-- removed -->
      </ds:SignedInfo>
      <ds:SignatureValue>
        <!-- removed -->
      </ds:SignatureValue>
    </ds:Signature>
  </saml2:Assertion>
</saml2p:Response>

```

```

    </ds:SignatureValue>
    <ds:KeyInfo>
      <ds:X509Data>
        <ds:X509Certificate>
          <!-- removed -->
        </ds:X509Certificate>
      </ds:X509Data>
    </ds:KeyInfo>
  </ds:Signature>
  <saml2:Subject>
    <saml2:NameID Format="urn:oasis:names:tc:SAML:2.0:nameid-format:persistent">4f3c1d1d-4532-4fe5-be35-
f0ee1c5722c0</saml2:NameID>
    <saml2:SubjectConfirmation Method="urn:oasis:names:tc:SAML:2.0:cm:bearer">
      <saml2:SubjectConfirmationData InResponseTo="ARQff9160d-8db6-4838-86cf-2a41a5808b48"
        NotOnOrAfter="2023-08-29T07:57:10.331Z"
        Recipient="https://samlsp.egov.sample.gov/login/saml2/sso/agovidp"/>
    </saml2:SubjectConfirmation>
  </saml2:Subject>
  <saml2:Conditions NotBefore="2023-08-29T07:56:40.331Z"
    NotOnOrAfter="2023-08-29T11:56:40.331Z">
    <saml2:AudienceRestriction>
      <saml2:Audience>https://samlsp.egov.sample.gov/saml2/service-provider-metadata/agovidp</saml2:Audience>
    </saml2:AudienceRestriction>
  </saml2:Conditions>
  <saml2:AttributeStatement>
    <saml2:Attribute Name="http://schemas.agov.ch/ws/2023/05/identity/claims/qa/verificationMethod">
      <saml2:AttributeValue>Counter</saml2:AttributeValue>
    </saml2:Attribute>
    <saml2:Attribute Name="http://schemas.agov.ch/ws/2023/05/identity/claims/qa/dateOfVerification">
      <saml2:AttributeValue>2023-10-11T00:00:00.000Z</saml2:AttributeValue>
    </saml2:Attribute>
    <saml2:Attribute Name="http://schemas.agov.ch/ws/2023/05/identity/claims/qa/validTillDate">
      <saml2:AttributeValue>2028-10-10T23:59:59.000Z</saml2:AttributeValue>
    </saml2:Attribute>
    <saml2:Attribute Name="http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress">
      <saml2:AttributeValue>wilhelm.tell@adnovum.ch</saml2:AttributeValue>
    </saml2:Attribute>
    <saml2:Attribute Name="http://schemas.xmlsoap.org/ws/2005/05/identity/claims/givenname">
      <saml2:AttributeValue>Wilhelm Friedrich</saml2:AttributeValue>
    </saml2:Attribute>
  </saml2:AttributeStatement>

```

```

</saml2:Attribute>
<saml2:Attribute Name="http://schemas.xmlsoap.org/ws/2005/05/identity/claims/surname">
  <saml2:AttributeValue>Tell</saml2:AttributeValue>
</saml2:Attribute>
<saml2:Attribute Name="http://schemas.agov.ch/ws/2023/05/identity/claims/languageOfCorrespondance">
  <saml2:AttributeValue>de</saml2:AttributeValue>
</saml2:Attribute>
<saml2:Attribute Name="http://schemas.agov.ch/ws/2023/05/identity/claims/socialSecurityNumber">
  <saml2:AttributeValue>7561111599997</saml2:AttributeValue>
</saml2:Attribute>
<saml2:Attribute Name="http://schemas.agov.ch/ws/2025/07/identity/claims/op/conversationId">
  <saml2:AttributeValue>a7243bf16926718408a92efe9d27774f</saml2:AttributeValue>
</saml2:Attribute>
<saml2:Attribute Name="http://schemas.agov.ch/ws/2023/05/identity/claims/dateOfBirth">
  <saml2:AttributeValue>1999-09-09</saml2:AttributeValue>
</saml2:Attribute>
<saml2:Attribute Name="http://schemas.agov.ch/ws/2023/05/identity/claims/nationality">
  <saml2:AttributeValue>CH</saml2:AttributeValue>
</saml2:Attribute>
<saml2:Attribute Name="http://schemas.agov.ch/ws/2023/05/identity/claims/placeOfBirth">
  <saml2:AttributeValue>Altdorf</saml2:AttributeValue>
</saml2:Attribute>
<saml2:Attribute Name="http://schemas.agov.ch/ws/2023/05/identity/claims/sex">
  <saml2:AttributeValue>l</saml2:AttributeValue>
</saml2:Attribute>
<saml2:Attribute Name="http://schemas.agov.ch/ws/2023/08/identity/claims/address/street">
  <saml2:AttributeValue>In der Burg</saml2:AttributeValue>
</saml2:Attribute>
<saml2:Attribute Name="http://schemas.agov.ch/ws/2023/08/identity/claims/address/houseNumber">
  <saml2:AttributeValue>1b</saml2:AttributeValue>
</saml2:Attribute>
<saml2:Attribute Name="http://schemas.agov.ch/ws/2023/08/identity/claims/address/zipCode">
  <saml2:AttributeValue>6403</saml2:AttributeValue>
</saml2:Attribute>
<saml2:Attribute Name="http://schemas.agov.ch/ws/2023/08/identity/claims/address/town">
  <saml2:AttributeValue>Küssnacht</saml2:AttributeValue>
</saml2:Attribute>
<saml2:Attribute Name="http://schemas.agov.ch/ws/2024/02/identity/claims/address/country">
  <saml2:AttributeValue>CH</saml2:AttributeValue>

```

```

    </saml2:Attribute>
    <saml2:Attribute Name="http://schemas.agov.ch/ws/2024/02/identity/claims/address/countryName">
      <saml2:AttributeValue>Schweiz</saml2:AttributeValue>
    </saml2:Attribute>
    <saml2:Attribute Name="http://schemas.agov.ch/ws/2024/02/identity/claims/address/qa/verificationMethod">
      <saml2:AttributeValue>SimpleLetter</saml2:AttributeValue>
    </saml2:Attribute>
  </saml2:AttributeStatement>
  <saml2:AuthnStatement AuthnInstant="2023-08-29T07:56:40.331Z"
    SessionIndex="_4957f0791b64f465e69699d551c747e7">
    <saml2:AuthnContext>
      <saml2:AuthnContextClassRef>urn:qa.agov.ch:names:tc:ac:classes:400</saml2:AuthnContextClassRef>
    </saml2:AuthnContext>
  </saml2:AuthnStatement>
</saml2:Assertion>
</saml2p:Response>

```

6.2.2 OIDC

The eGov application sent the following Authorization Request as a http redirect over the user's webbrowser to the authorization endpoint of AGOV IdP :

```

GET https://idp.agov.admin.ch/oidc/auth?
  client_id=https%3A%2F%2Foidc.egov.sample.gov%2F
  &response_type=code
  &redirect_uri=https%3A%2F%2Foidc.egov.sample.gov%2Ftest%2Fapi%2Flogin%2Foauth2%2Fcode%2Fatb
  &scope=openid%20email%20profile%20svnr%20address%20agovProfile
  &response_type=code
  &code_challenge_method=S256
  &code_challenge=TFeGQX0pYFd-pi5Pe8JDqwlpl_27Vise-GXca5HJtdM
  &state=uqf5xpIi0q
  &nonce=f3a51w4zqpm

```

i Note: the eGov application was registered with AGOV connect and declared to need the address and svnr, and set a default of 300 for the required AGOVaq level.

The AGOV IdP validated the requested, authenticated the user with the AGOV account ID "4f3c1d1d-4532-4fe5-be35-f0ee1c5722c0" and returned a https 302 status with the following redirect location:

```
https://oidc.egov.sample.gov/test/api/login/oauth2/code/atb?  
code=9jDg-csn2mEjgeapcQcprYBDaMXY097zHHO2QnE5XwF2i8S03zbtbR7Ls9hYsKt-t_DVkJDurOSUXpn_uU1yvcU_NXLBewq1F-  
Fg684FXEWZe5qeh0FBPrUeC2I3TZ2Uh  
&state=uqf5xp1i0q
```

The eGov application calls again the IdP to fetch the ID Token with the following POST body:

```
client_id: https://oidc.egov.sample.gov/  
code: 9jDg-csn2mEjgeapcQcprYBDaMXY097zHHO2QnE5XwF2i8S03zbtbR7Ls9hYsKt-t_DVkJDurOSUXpn_uU1yvcU_NXLBewq1F-  
Fg684FXEWZe5qeh0FBPrUeC2I3TZ2Uh  
code_verifier: bY3sDjL8KI8izWHOT3lrKjXIGw4qTIJXgfPDKVB7kWP  
grant_type: authorization_code  
redirect_uri: https://oidc.egov.sample.gov/test/api/login/oauth2/code/atb
```

The received answer contained the following ID Token:

```

{
  "sub": "4f3c1d1d-4532-4fe5-be35-f0ee1c5722c0",
  "qa_DateOfVerification": 1693267200,
  "birthdate": "1999-09-09",
  "gender": "male",
  "socialSecurityNumber": 7561111599997,
  "iss": "https://idp.agov.admin.ch/",
  "language": "de",
  "locale": "de-CH",
  "acr": "urn:qa.agov.ch:names:tc:ac:classes:400",
  "azp": "https://oidc.egov.sample.gov/",
  "auth_time": 1693294904,
  "exp": 1693309376,
  "iat": 1693294976,
  "jti": "a0337203-7e4a-4d08-ae4f-9c8660c0d736",
  "email": "wilhelm.tell@saga.ch",
  "address": {
    "street_address": "In der Burg 1b",
    "country": "Schweiz",
    "street": "In der Burg",
    "countryCode": "CH",
    "formatted": "In der Burg 1b\nCH-6403 Küssnacht",
    "locality": "Küssnacht",
    "verificationMethod": "SimpleLetter"
    "verified": true,
    "house_number": "1b",
    "postal_code": "6403"
  },
  "conversationId": "4eb0015b35095df7f3b58f1f59651d33",
  "placeOfBirth": "Altdorf",
  "email_verified": true,
  "given_name": "Wilhelm Friedrich",
  "nonce": "f3a51w4zqpm",
  "aud": "https://oidc.egov.sample.gov/",
  "nationality": "CH",
  "qa_verificationMethod": "Bmid",
  "qa_validTillDate": 1851206399,
  "family_name": "Tell"
}

```

