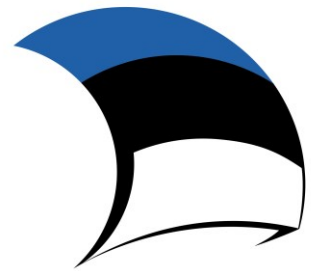
REPUBLIC OF ESTONIA
INFORMATION SYSTEM AUTHORITY

Web service integration with State Single Sign-On (GOVSSO)

Helen Raamat (Information System Authority)
Aare Nurm (Nortal AS)
Alar Kvell (Nortal AS)
07.06.2022

European Union
European Regional
Development Fund



Investing
in your future

Agenda

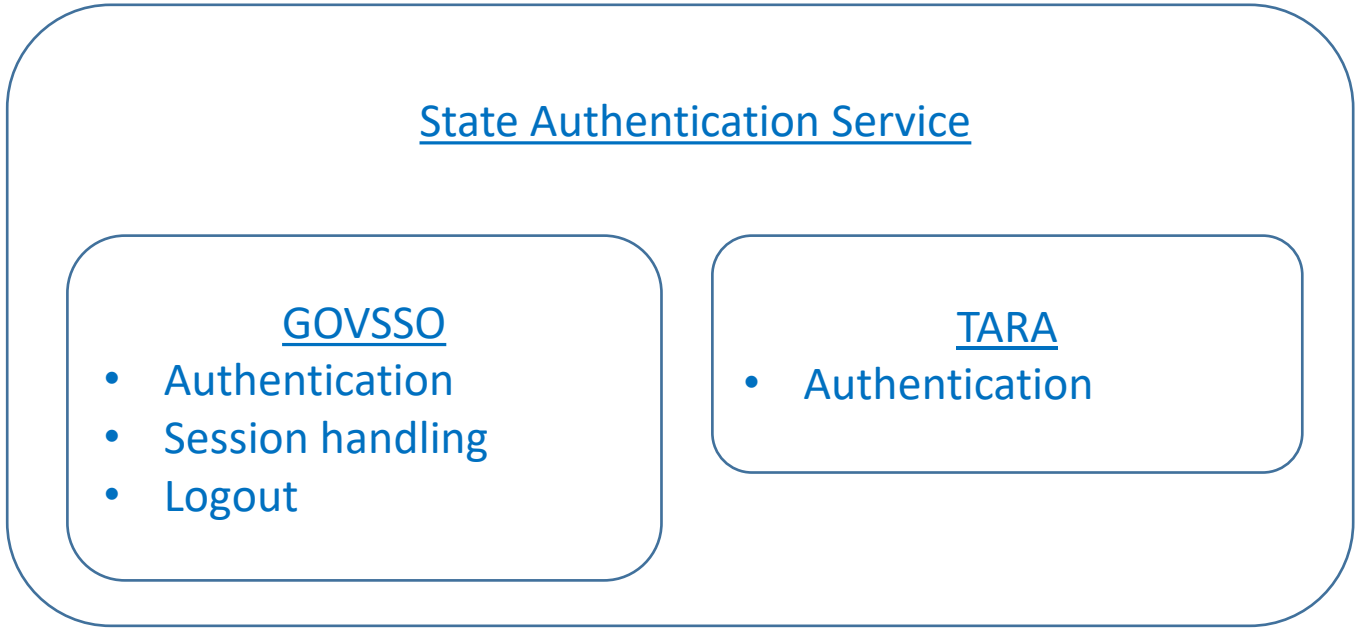
- Introduction
- DEMO/PROD environment application process
- Q&A

- Authentication flows
- API parameters, required validations
- Testing
- Q&A

What is GOVSSO?

- Purpose
  - To optimize cost of authentication by minimizing the usage of billable services (Mobile-ID; Smart-ID; OCSP for ID-card).
  - To give better user experience when moving between different state services.

What is GOVSSO?

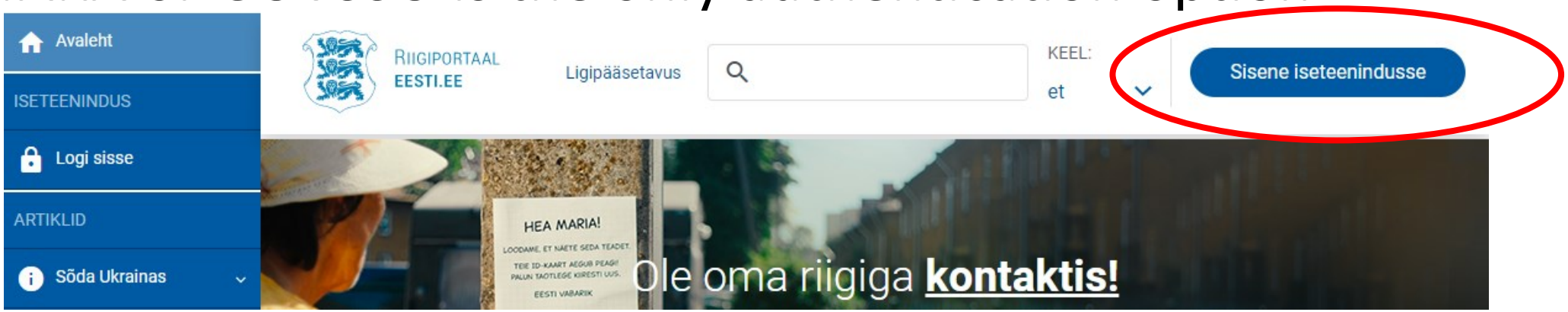


What is GOVSSO?

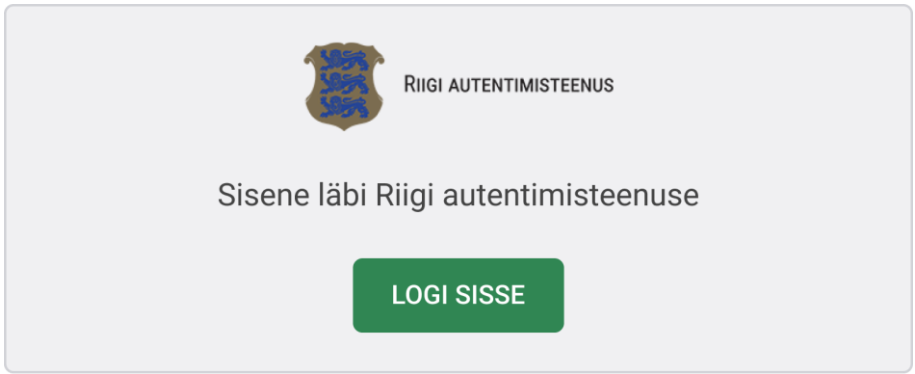
- Users should not differentiate when system is moved from TARA to GOVSSO
- Same Login button recommendations as for TARA

UI recommendation for login

- If TARA or GOVSSO is the only authentication option



- If there are several authentication options:



UI recommendation for logout

- Reference what service has been logged out of
- Avoid instructing to close all windows

GOVSSO roadmap

- 03.2022 – 06.2022 GOVSSO stage 2
  - Feedback, refinement
  - GOVSSO software ready for production
- 08.2022 GOVSSO as production service

Application process for DEMO and PROD environments

- Register your e-service in our demo environment and conduct integration tests.
- Register your e-service in our production environment.
- The application forms will become available on RIA website.
- Submit your completed and signed applications at help@ria.ee.
  - The application must be signed by the legal representative of your organization.
- After registration you will receive required credentials to use GOVSSO services.

Considerations on selecting values

- Client application name
  - Should be recognized by end user. Avoid technical terms.
- Client application short name
  - Should be recognized by end user. Should be relatable with application name.
- Client application logo
  - SVG format

Considerations on selecting values

- Authentication return URL-s
  - One or multiple URL-s
- Logout URL-s
  - One or multiple URL-s
- Back-channel logout URL
  - Recommended to limit access to only RIA IP-address

Additional information

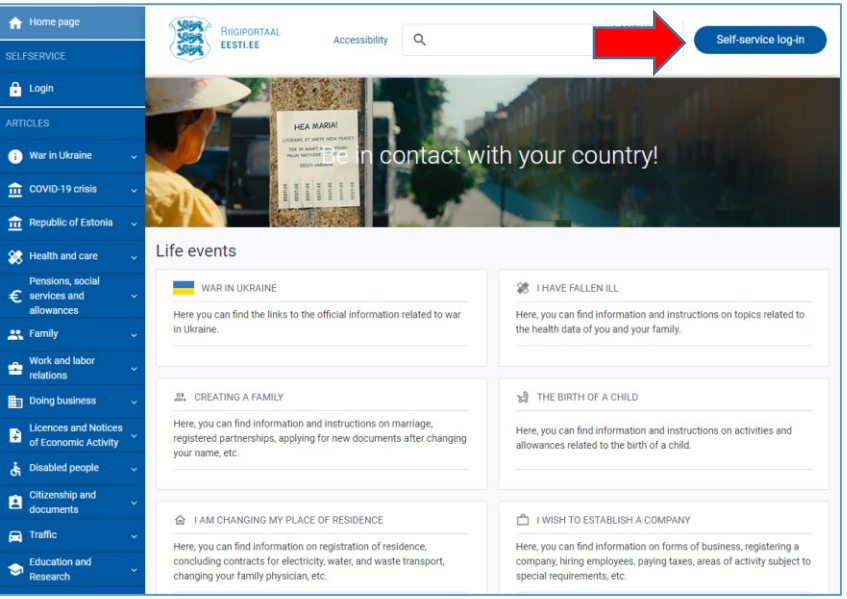
- https://e-gov.github.io/GOVSSO/
  - This page is the main source for technical information

Questions and answers

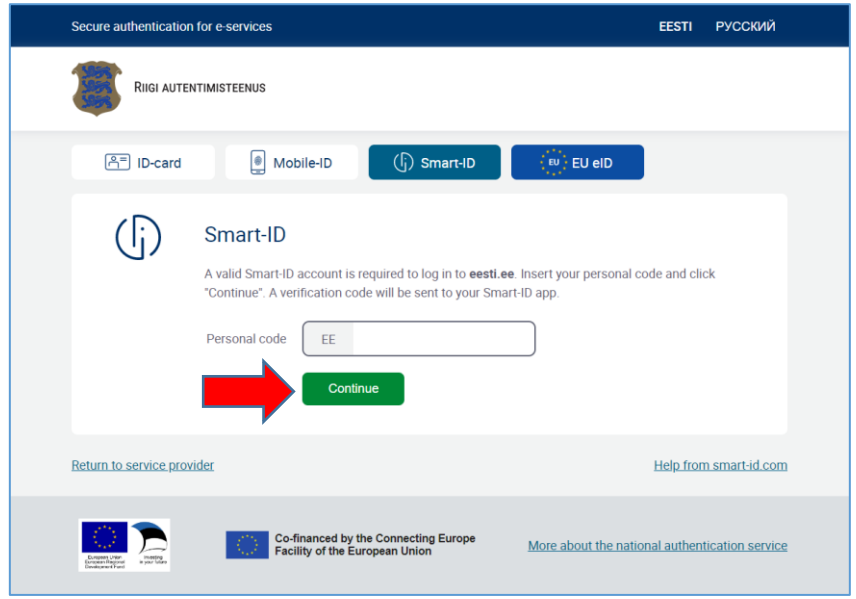
- Questions?

Usage flow without GOVSSO

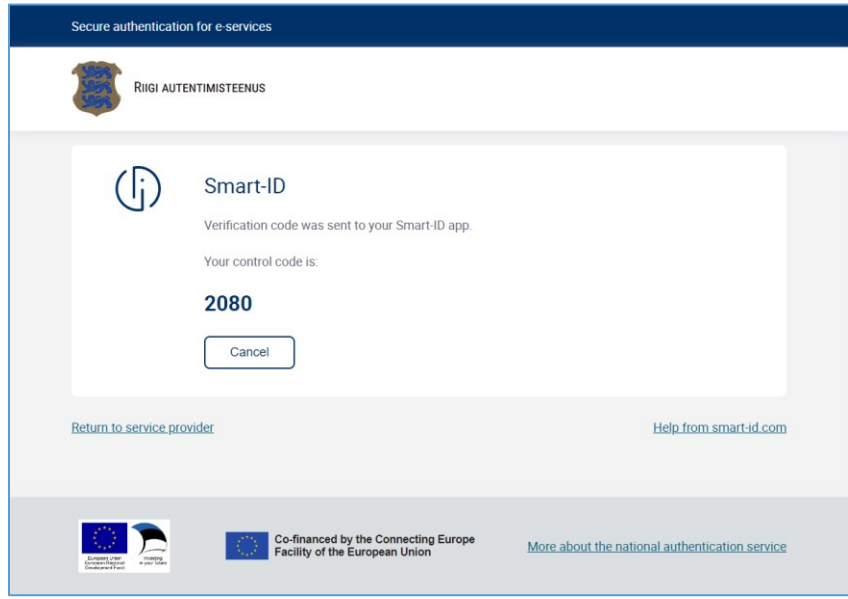
1



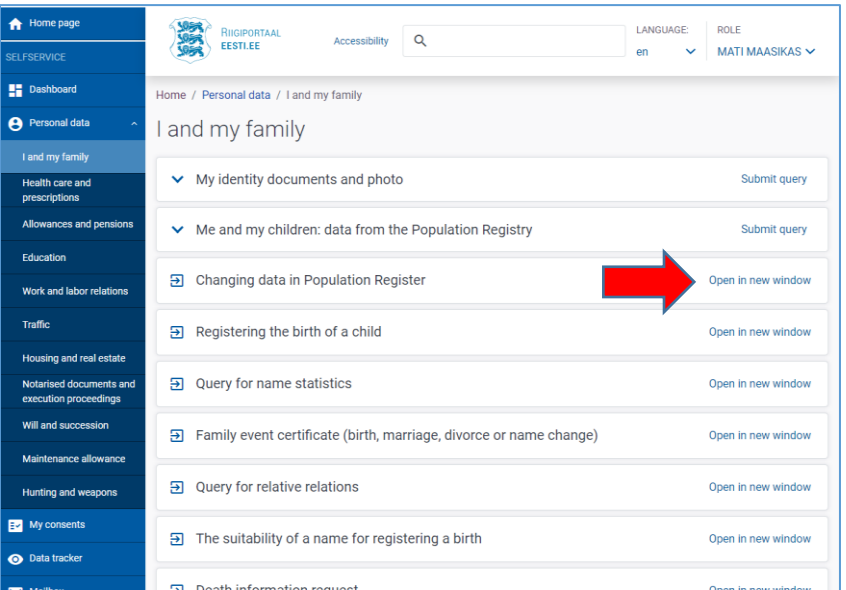
2



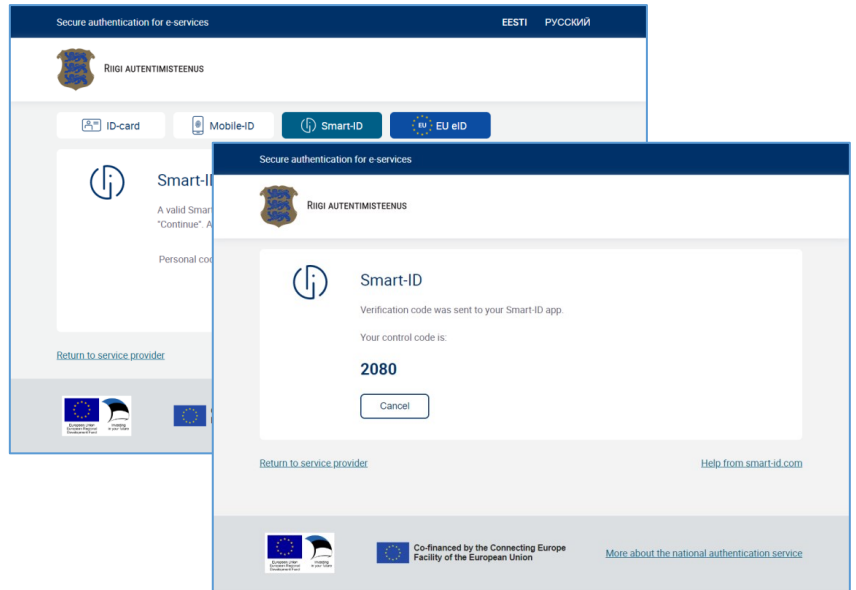
3



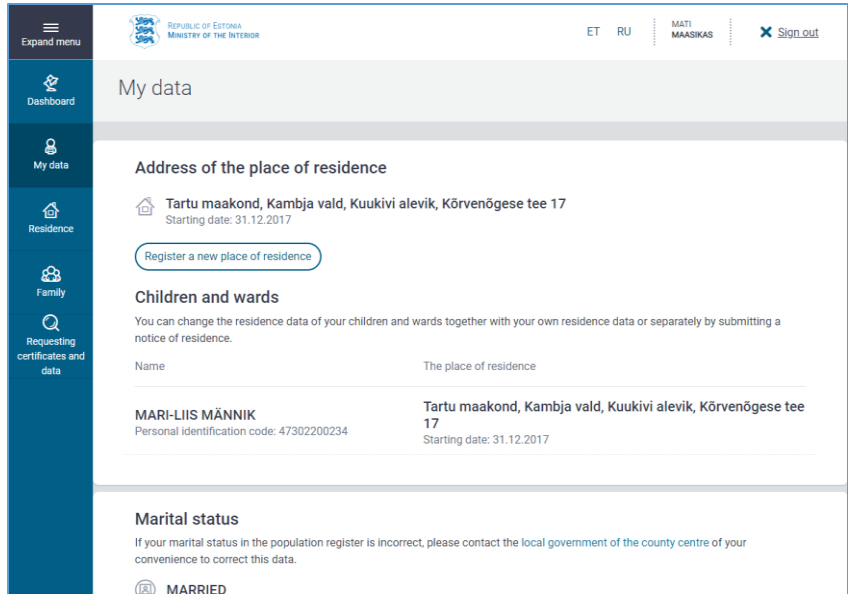
4



5 & 6

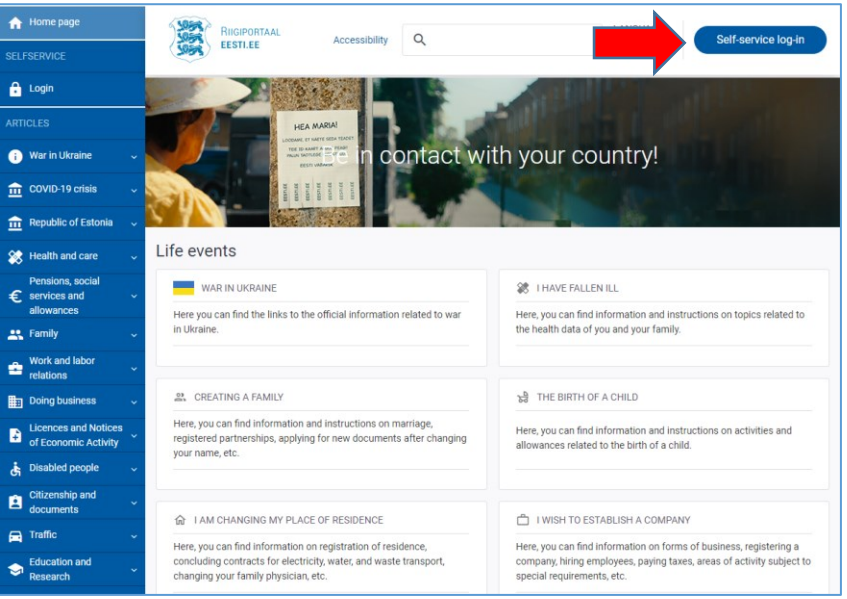


7

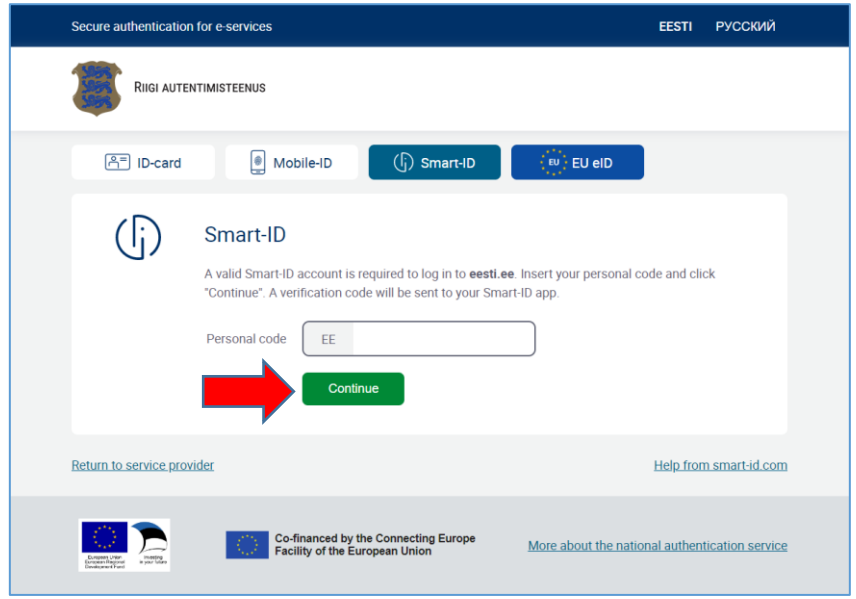


Usage flow with GOVSSO

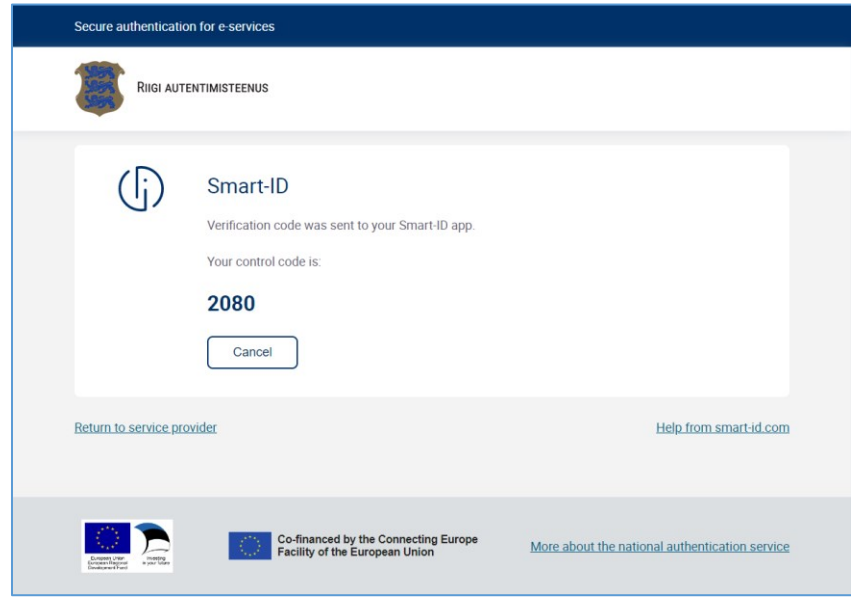
1



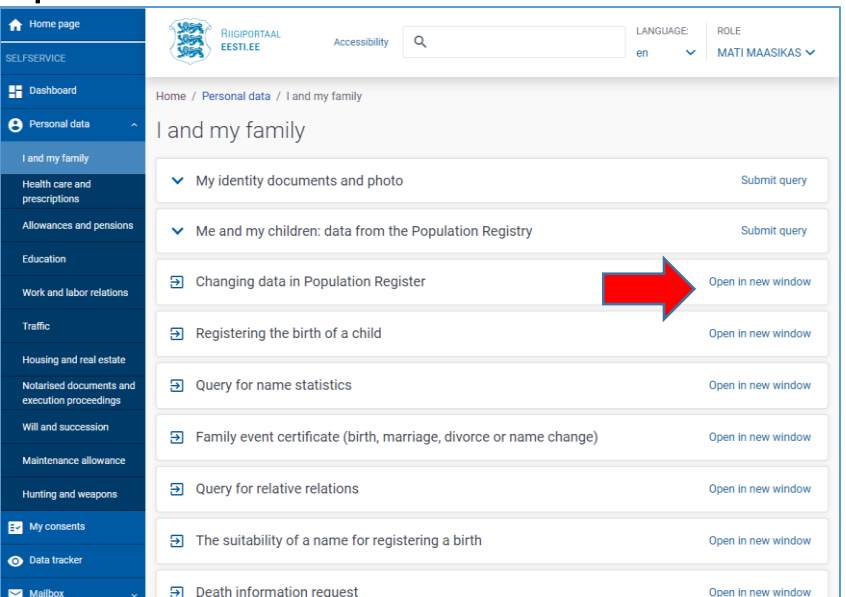
2



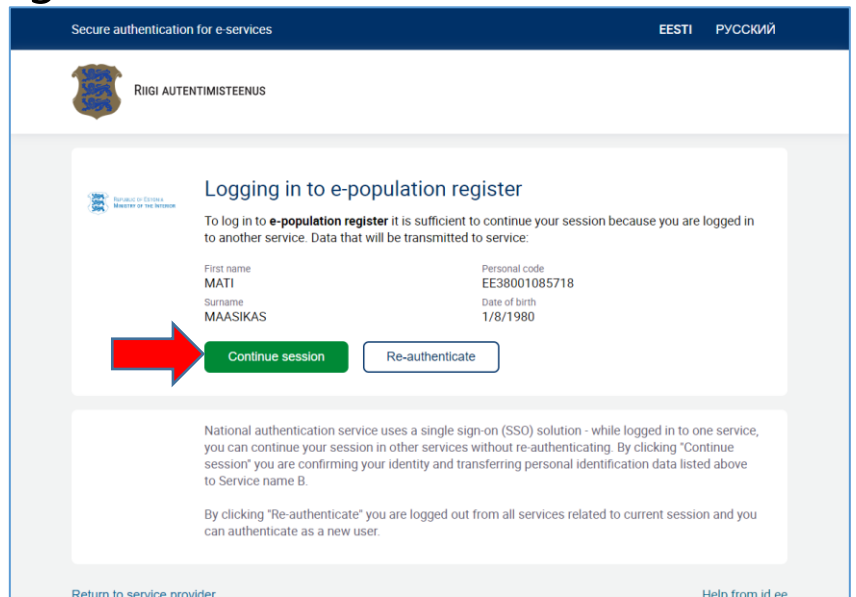
3



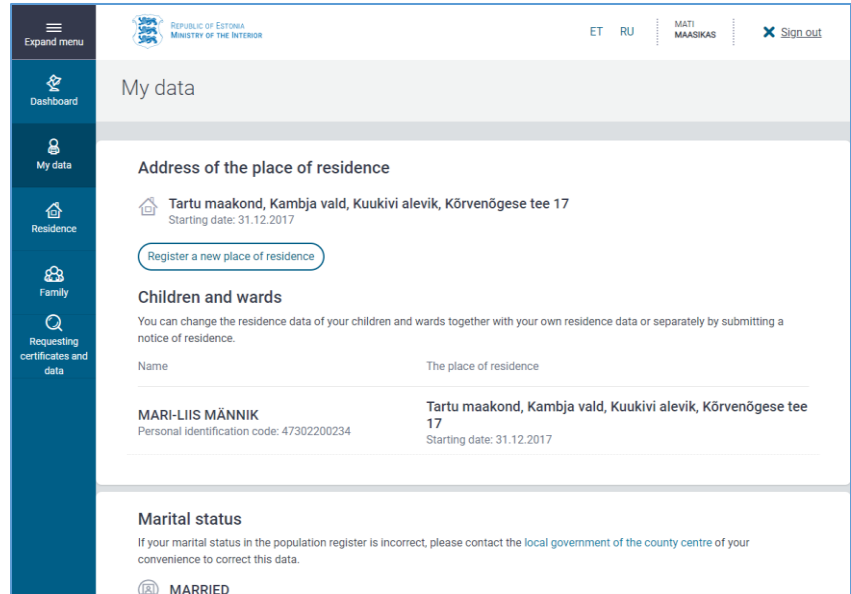
4



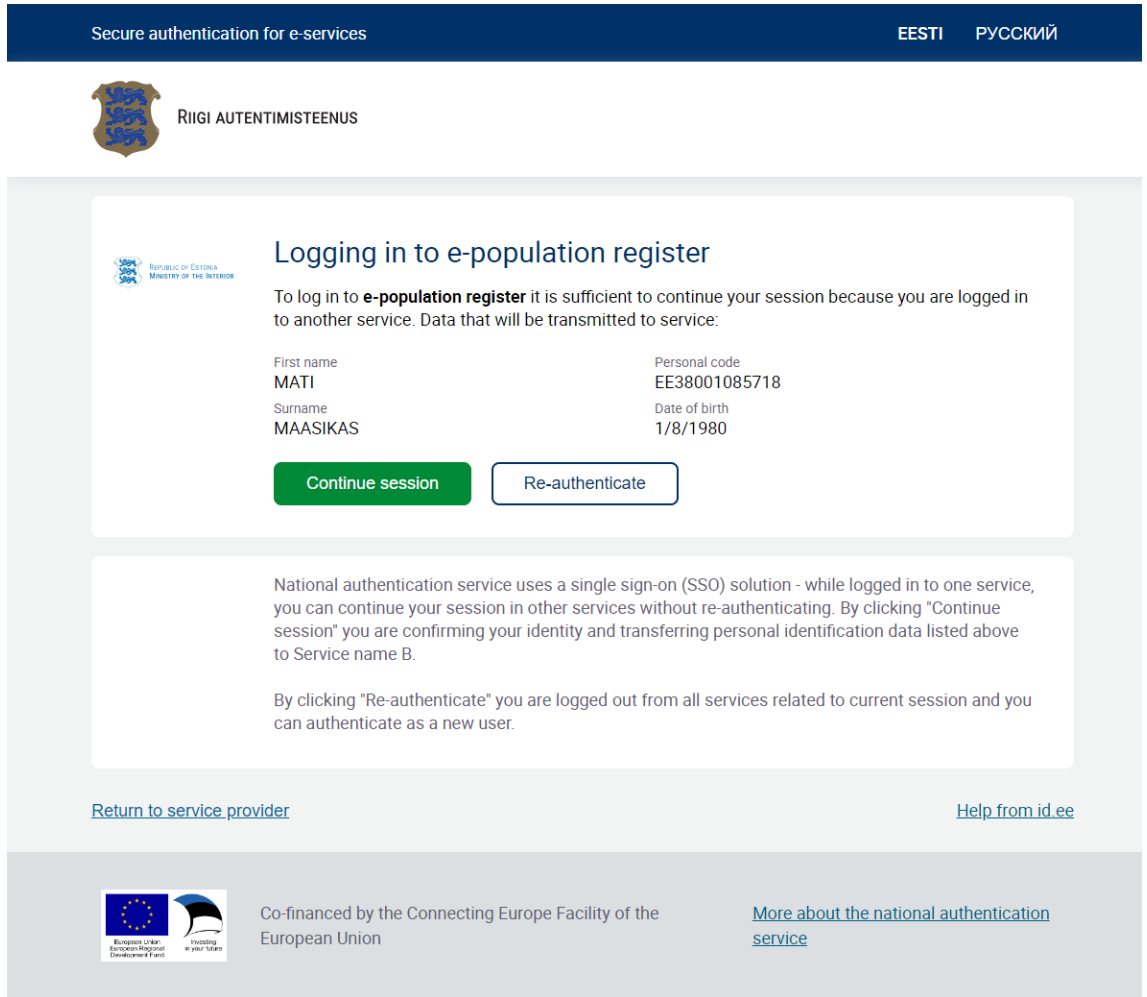
5



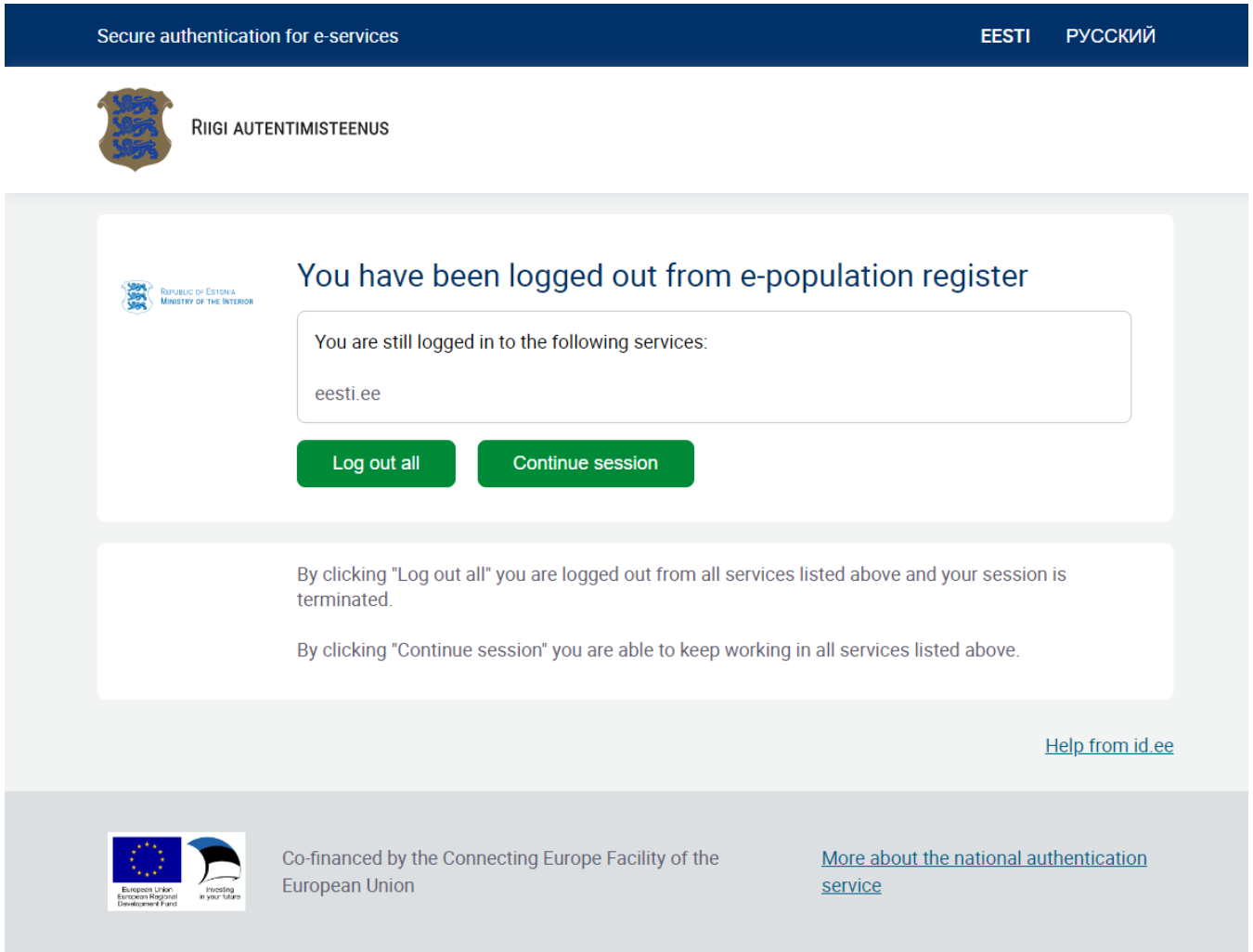
6



GOVSSO session continuation



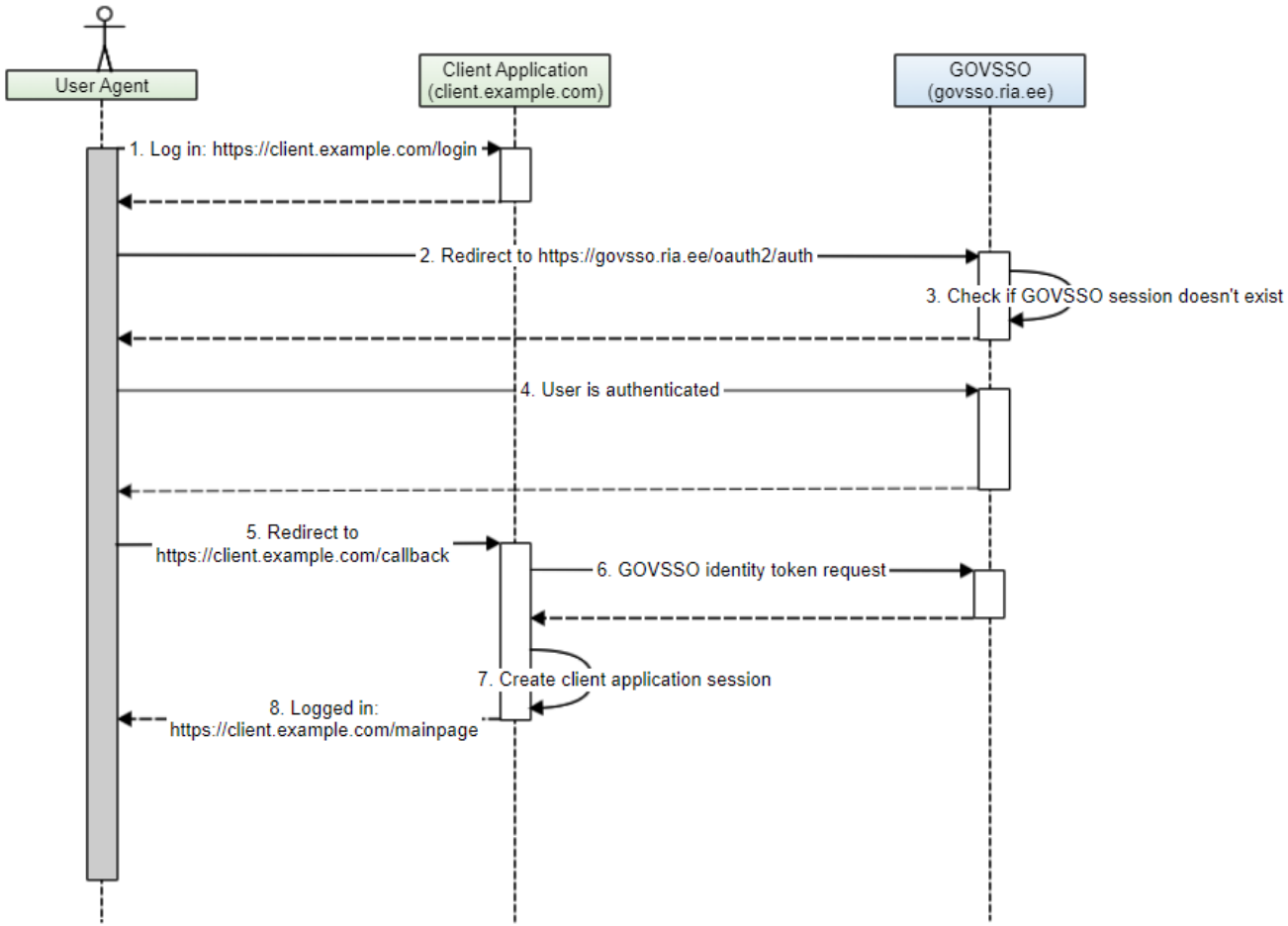
GOVSSO logout



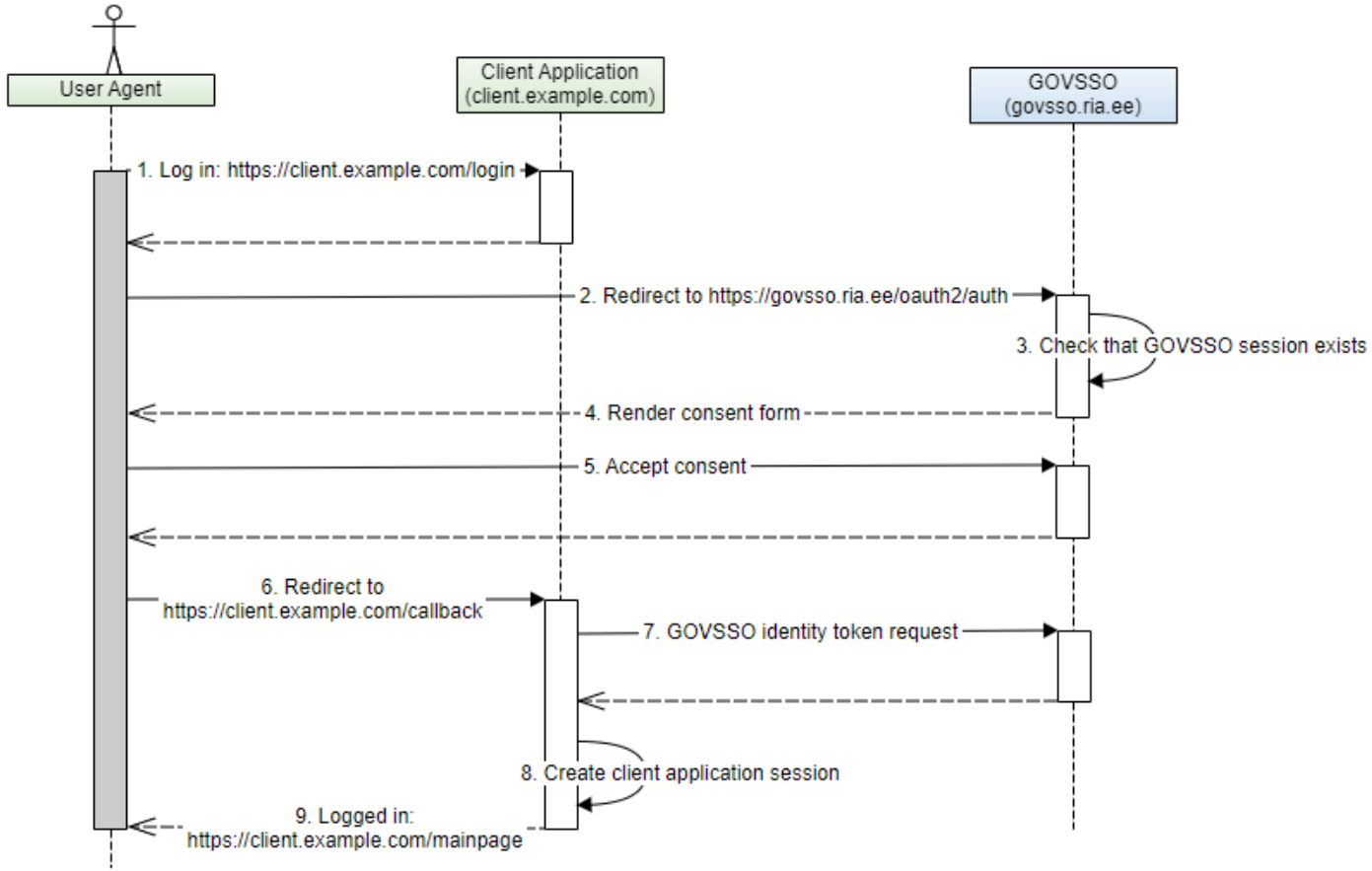
GOVSSO flow demonstration

- Authentication
  - Single service
  - Two services
- Session update
- Logout
  - One service
  - Logout all

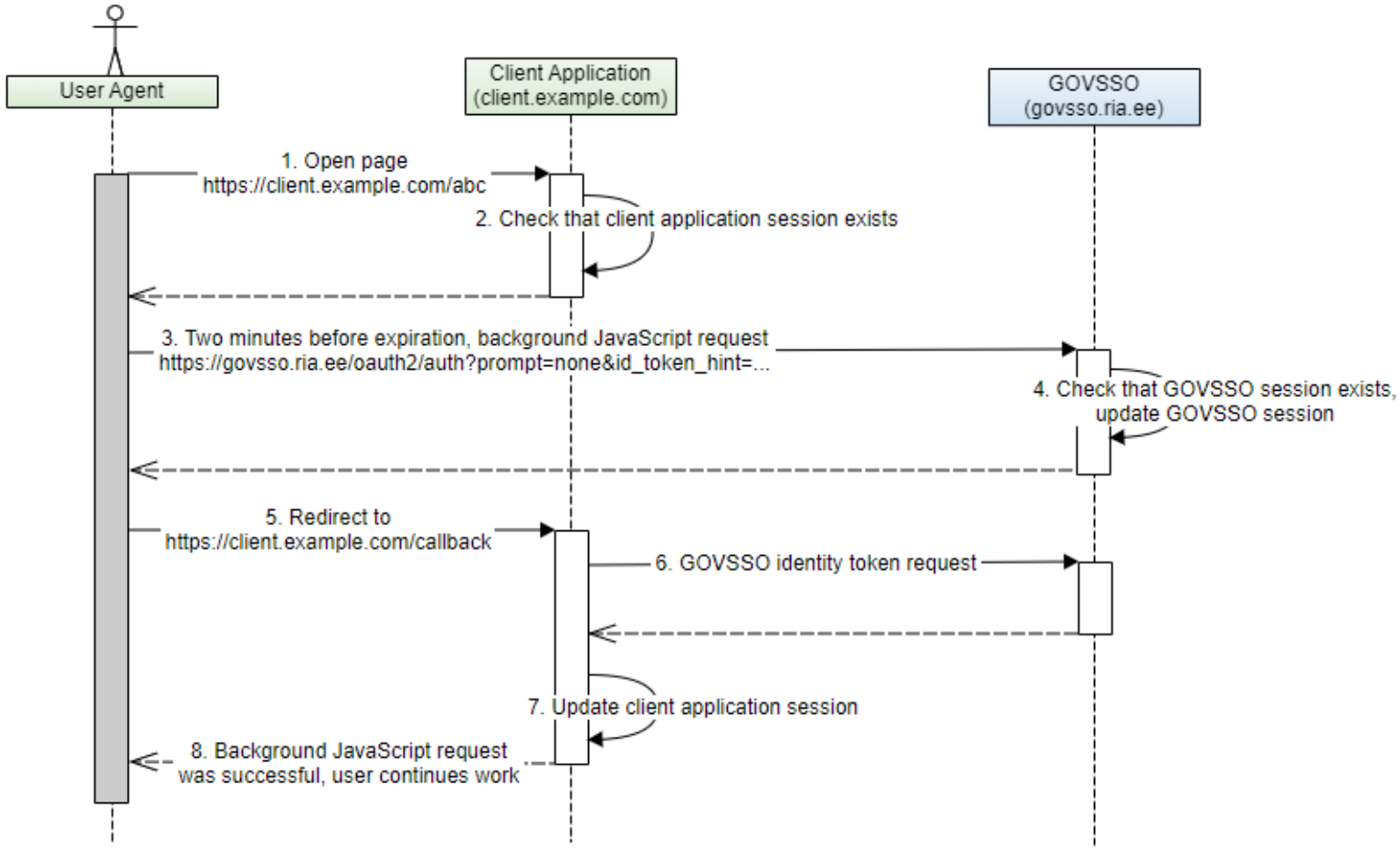
GOVSSO authentication flow: new session



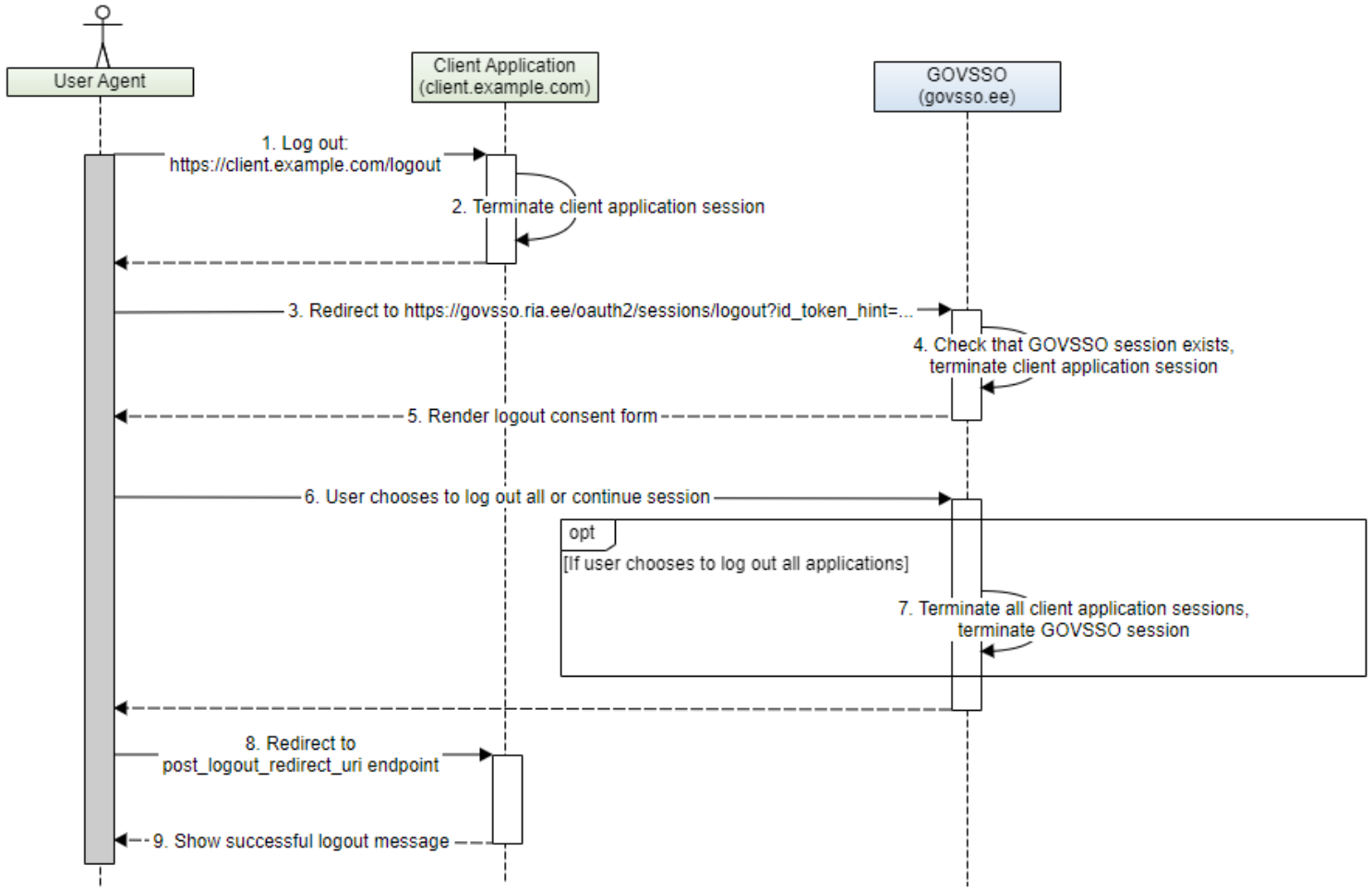
GOVSSO authentication flow: session exists



GOVSSO session update



GOVSSO logout



/.well-known/openid-configuration endpoint


```
{
    "issuer": "https://govsso-demo.ria.ee/",
    "authorization_endpoint": "https://govsso-demo.ria.ee/oauth2/auth",
    "token_endpoint": "https://govsso-demo.ria.ee/oauth2/token",
    "jwks_uri": "https://govsso-demo.ria.ee/.well-known/jwks.json",
    "subject_types_supported": [
        "public"
    ],
    "response_types_supported": [
        "code"
    ],
    "claims_supported": [
        "sub",
        "acr",
        "amr",
        "at_hash",
        "aud",
        "auth_time",
        "exp",
        "iat",
        "iss",
        "jti",
        "nonce",
        "birthdate",
        "family_name",
        "given_name",
        "sid"
    ],
```

/.well-known/openid-configuration endpoint


```
"grant_types_supported": [
    "authorization_code"
],
"response_modes_supported": [
    "query"
],
"scopes_supported": [
    "openid"
],
"token_endpoint_auth_methods_supported": [
    "client_secret_basic"
],
"id_token_signing_alg_values_supported": [
    "RS256"
],
"request_uri_parameter_supported": false,
"claims_parameter_supported": false,
"backchannel_logout_supported": true,
"backchannel_logout_session_supported": true,
"ui_locales_supported": [
    "et",
    "en",
    "ru"
],
```

/.well-known/openid-configuration endpoint


```
    "acr_values_supported": [
        "low",
        "substantial",
        "high"
    ],
    "claim_types_supported": [
        "normal"
    ],
    "service_documentation": "https://e-gov.github.io/GOVSSO/",
    "end_session_endpoint": "https://govsso-demo.ria.ee/oauth2/sessions/logout"
}
```

/.well-known/jwks endpoint


```
{
    "keys": [
        {
            "use": "sig",
            "kty": "RSA",
            "kid": "public:4a91f86b-340c-4c3a-a3c1-204a74479e1d",
            "alg": "RS256",
            "n":
                "sdvigkdkxbAvzh-GfgdNz4iOf8gUTKHJTLhJLEV7TOiCsw5j4Rn0AO497dm8-UXFhs76ArIAoh5pbeivBOwv9dPfvQaskFhbM
                YeFvwtjucH0OIAeo7Qy69ixA2tb3fibiNsXnpy3RvCpe5yBP5fLsRpJGJnezp7OwjeXEZchq5Y43U0dIXFXMdAlAAUJ6pQ-hX4
                qcUU3sEuwLLDOLwe0EerXiKBhd15CPnubHLcqrzcz_BJov1AOWhkfKMqjIQDPmSRO9lQOLjsp2W9_pA-q1ZxsyRKkhCmofcHeT
                U8RoPFaiCkpy-HXn-fQD7EVsk25s-1zyBovd2ZKO3jKDkr0JXR2llZgZQNU5GGKPCN_Xft4A5uF1tfjx_EATQG8OThYDM",
            "e": "AQAB"
        }
    ]
}
```

Key usage

- Client application should buffer the public key (it needs to be buffered together with kid value). It needs to compare the kid value from JWT header with buffered kid value.

HEADER:


```
{
  "alg": "RS256",
  "kid": "public:4a91f86b-340c-4c3a-a3c1-204a74449e1d",
  "typ": "JWT"
}
```

Key usage

- If key values from JWT header and buffer match, buffered key can be used. If not, client application needs to make request to public signature key endpoint and select key corresponding to kid value received from JWT header and buffer it.

GOVSSO flow: Authentication request

```
GET https://govsso.ria.ee/oauth2/auth?

redirect_uri=https%3A%2F%2Fclient.example.com%2Fcallback&
scope=openid&
state=hkMVY7vjuN7xyL15&
response_type=code&
client_id=58e7ba35aab5b4f1671a&
ui_locales=en&
nonce=fsdsfwrerhtry3qeewq&
acr_values=substantial
```

GOVSSO flow: Authentication request

URL element	*compulsory*	*example*	*explanation*
protocol, host, port and path	yes	`https://govsso.ria.ee/oauth2/auth`	`/oauth2/auth` is the OpenID Connect-based authentication endpoint of the GOVSSO service (the concept of 'authorization' originates from the OAuth 2.0 standard protocol). The URL is provided from OIDC server public discovery service: `https://govsso.ria.ee/.well-known/openid-configuration` as `authorization_endpoint` parameter.
client_id	yes	`client_id=58e7ba35aab5b4f1671a`	Client identifier. The client identifier is issued by RIA upon registration of the client application as a user of the authentication service.
redirect_uri	yes	`redirect_uri=https%3A%2F%2Fclient.example.com%2Fcallback`	Redirect URL ([OAUTH] "3.1.2. Redirection Endpoint"). The redirect URL is selected by the institution. The redirect URL may include the query component. URL encoding should be used, if necessary [URLENC]. It is not permitted to use the URI fragment component (`#` and the following component; [URI] "3.5. Fragment"). The URL protocol, host, port and path must match one of the pre-registered redirect URLs of given client application registration metadata (see `client_id` parameter).

GOVSSO flow: Authentication request

scope	yes	`scope=openid`	The authentication scope. Space delimited list of requested scopes.
			
			`openid` scope is compulsory to signal that this is an OIDC authentication request.
			In the default `scope` of openid GOVSSO will issue ID Tokens with the following attributes:
			
			`sub` (physical person identifier)
			`given_name`
			`family_name`
			`birthdate`
			
			Presence of given attribute values will depend on the amount of information that is returned within TARA ID Tokens.

state	yes	`state=hkMVY7vjuN7xyL15`	Security code against false request attacks (cross-site request forgery, CSRF). Read more about formation and verification of state under “7.3 Protection against false request attacks”.
response_type	yes	`response_type=code`	Determines the manner of communication of the authentication result to the server. Only value `code` is allowed as only authorization code flow is supported by GOVSSO

GOVSSO flow: Authentication request

ui_locales	no	`ui_locales=et`	Selection of the user interface language. The following languages are supported: `et`, `en`, `ru`. By default, the user interface is in Estonian language. The client can select the desired language. This will also set the GUI language for TARA service views.
nonce	no	`nonce=fsdsfwrerhtry3qeewq`	A unique parameter which helps to prevent replay attacks based on the OIDC protocol ([OIDC-CORE] "3.1.2.1. Authentication Request").
acr_values	no	`acr_values=substantial`	The minimum required level of authentication based on the eIDAS level of assurance (LoA). Allowed values are: `low`, `substantial`, `high`. `high` is used by default if the value has not been set in request. GOVSSO will store the authentication level of assurance in the SSO session object as an Authentication Context Class Reference (`acr`) claim, from TARA ID Token response. Upon each GOVSSO authentication request, GOVSSO will check that the requested level of assurance (`acr_values` parameter value) is lower or equal to the `acr` claim value of the GOVSSO session. If the SSO session `acr` value (level of assurance) is lower than requested, the previous GOVSSO session is automatically terminated and a new authentication is requested from TARA. After successful authentication a new SSO session is created.

GOVSSO flow: Authentication response

- Sent request

```
GET https://govsso.ria.ee/oauth2/auth?

redirect_uri=https%3A%2F%2Fclient.example.com%2Fcallback&
scope=openid&
state=hkMVY7vjuN7xyL15&
response_type=code&
client_id=58e7ba35aab5b4f1671a&
ui_locales=en&
nonce=fsdsfwrerhtry3qeewq&
acr_values=substantial
```

- „Response“ is returned through HTTP 302 redirect

- Received response

```
HTTP GET https://client.example.com/callback?

code=71ed5797c3d957817d31&
state=hkMVY7vjuN7xyL15
```

GOVSSO flow: Authentication response

URL element	*example*	*explanation*
protocol, host, port and path	`https://client.example.com/callback`	Matches the `redirect_uri` value sent in the authentication request.
code	`code=71ed579...`	The authorization code to request the ID Token.
state	`state=hkMVY7vjuN7xyL15`	Security code against false request attacks. The security code received in the authentication request is mirrored back. Read more about forming and verifying `state` from 'Protection against false request attacks'.

GOVSSO flow: Authentication error response

```
HTTP GET https://client.example.com/callback?

error=invalid_scope&
error_description=Required+scope+%3Copenid%3E+not+provided.+GOVSSO+does+not+allow+this+request+to+be+processed&
state=hkMVY7vjuN7xyL15
```

- If user cancels authentication in GOVSSO same mechanism is used to return the cancel action.
  - error=user_cancel
- Error description should be used only for debugging

GOVSSO flow: ID Token request

- HTTPS is obligatory and GOVSSO certificate (chain) must be trusted
- TLS 1.2 is used (TLS 1.3 will come in future)

GOVSSO flow: ID Token request

```
POST /oauth2/token HTTP/1.1
Host: govsso.ria.ee
Content-Type: application/x-www-form-urlencoded
Authorization: Basic czZCaGRSa3F0MzpnWDFmQmF0M2JW

grant_type=authorization_code&
code=Splxl0BeZQQYbYS6WxSbIA&
redirect_uri=https%3A%2F%2client.example.com%2Fcallback
```

- The body of the HTTP POST request must be presented in a serialized format based on the OpenID Connect protocol.
- ID Token must be requested within 30 seconds after GOVSSO has issued authorization code

GOVSSO flow: ID Token request authentication

- Request must include the Authorization request header
- The value is formed of the word Basic, a space, and a string <client_id>:<client_secret> encoded in the Base64 format.

GOVSSO flow: ID Token request

Parameter	*parameter type*	*example*	*explanation*
protocol, host, port and path	query	`https://govsso.ria.ee/oauth2/token`	GOVSSO server token endpoint URL. Published in GOVSSO discovery endpoint `token_endpoint` parameter value.
grant_type	body	`grant_type=authorization_code`	The `authorization_code` value required based on the protocol. [OIDC-CORE] "3.1.3.1. Token Request"
code	body	`code=SplxlOBeZQQYbYS6WxSbIA`	The authorization code received from the authentication service.
redirect_uri	body	`redirect_uri=https%3A%2F%2client.example.com%2Fcallback`	The redirect URL sent in the authentication request.

GOVSSO flow: ID Token response

- Sent request

```
POST /oauth2/token HTTP/1.1
Host: govsso.ria.ee
Content-Type: application/x-www-form-urlencoded
Authorization: Basic czZCaGRSa3F0MzpnWDFmQmF0M2JW

grant_type=authorization_code&
code=Splxl0BeZQQYbYS6WxSbIA&
redirect_uri=https%3A%2F%2client.example.com%2Fcallback
```

- Received response

```
HTTP/1.1 200 OK
Content-Type: application/json
Cache-Control: no-store
Pragma: no-cache

{
 "access_token": "EKN-4fXC4n1RdkegKk-M0DRxZ8RwJYZ_EwW-9zLCYcA.7GT7Xq2deLvWzrrFq6f0DNwL6INW2PYRDPPEFMbws1o",
 "token_type": "Bearer",
 "expires_in": 3600,
 "id_token": "eyJhbGciOiJSUzI1NiIsImtpZCI6InB1YmxpYzoyMDQ0YmVlOC03MmY0LTQxYTMtYTRl
```

GOVSSO flow: ID Token response

Parameter	*explanation*
access_token	OAuth 2.0 access token. With the access token the client application can request authenticated user's data from userinfo endpoint. **Not used in GOVSSO because GOVSSO session management is purely ID Token dependent. All user data is already available in the ID Token.**
token_type	OAuth 2.0 access token type with `bearer` value. Not used in GOVSSO.
expires_in	The validity period of the OAuth 2.0 access token. Not used in GOVSSO.
id_token	ID Token, encapsulated in JWS Compact Serialization form ([JWS] chapter 3.1). The ID Token itself is issued in JSON Web Token [JWT] format .

- id_token is only value that should be used
- Unknown parameters must be ignored

ID Token structure


```
{
  "jti": "663a35d8-92ec-4a8d-95e7-fc6ca90ebda2",
  "iss": "https://govsso.ria.ee/",
  "aud": [
    "sso-client-1"
  ],
  "exp": 1591709871,
  "iat": 1591709811,
  "sub": "EE60001018800",
  "birthdate": "2000-01-01",
  "family_name": "O'CONNEŽ-ŠUSLIK TESTNUMBER",
  "given_name": "MARY ÄNN",
  "amr": [
    "mID"
  ],
  "acr": "high",
  "at_hash": "AKIDtvBT2JS_02tkl_DvuA",
  "nonce": "POYXXoyDo49deYC3o5_rG-ig3U4o-dtKgcym5SyHfCM",
  "sid": "f5ab396c-1490-4042-b073-ae8e003c7258",
}
```

ID Token structure

ID Token element (claim)	*example*	*explanation*
jti	`"jti":"663a35d8-92ec-4a8d-95e7-fc6ca90ebda2"`	ID Token unique identifier ([JWT] "4.1.7. jti (JWT ID) Claim").
iss	`"iss":"https://govsso.ria.ee/"`	Issuer Identifier, as specified in [OIDC-CORE].

aud	`"aud": [ "sso-client-1" ]`
	
	or
	
	`"aud": "sso-client-1"`
		Unique ID of a client application in GOVSSO client database. ID belongs to the client that requested authentication (the value of `client_id` field is specified in authentication request).
		
		String or array of strings. A single aud value is present in GOVSSO tokens.

exp	`"exp": 1591709871`	The expiration time of the ID Token (in Unix *epoch* format).
iat	`"iat": 1591709811`	The time of issue of the ID Token (in Unix *epoch* format).

ID Token structure

sub	`"sub": "EE60001018800"`	The identifier of the authenticated user (personal identification code or eIDAS identifier) with the prefix of the country code of the citizen (country codes based on the ISO 3166-1 alpha-2 standard). The subject identifier format is set by TARA authentication service ID Token [TARA] "4.3.1 Identity token". NB! in case of eIDAS authentication the maximum length is 256 characters.
birthdate	`"birthdate": "2000-01-01"`	The date of birth of the authenticated user in the ISO_8601 format. Only sent in the case of persons with Estonian personal identification code and in the case of eIDAS authentication.
given_name	`"given_name": "MARY ÄNN"`	The first name of the authenticated user (the test name was chosen because it consists special characters).
family_name	`"family_name": "O'CONNEŽ-ŠUSLIK TESTNUMBER"`	The surname of the authenticated user (the test name was selected because it includes special characters).
amr	`"amr": [ "mID" ]`	Authentication method reference. The authentication method used for user authentication. A single `amr` value is present in GOVSSO tokens. Possible values: `mID` - Mobile-ID `idcard` - Estonian ID card `eIDAS` - European cross-border authentication `smartid` - Smart-ID Available authentication methods depend on TARA authentication service and the list may be extended in the future [TARA] "4.1 Authentication request".

ID Token structure

nonce	`"nonce": "POYXXoyDo49deYC3o5_rG-ig3U4o-dtKgcym5SyHfCM"`	Security element. The authentication request's `nonce` parameter value. Value is present only in case the `nonce` parameter was sent in the authentication request.
acr	`"acr": "high"`	Authentication Context Class Reference. Signals the level of assurance of the authentication method that was used. Possible values: `low`, `substantial`, `high`. The element is not used if the level of assurance is not applicable or is unknown.
at_hash	`"at_hash": "AKIDtvBT2JS_02tkl_DvuA"`	The access token hash calculated as described in OIDC specification [OIDC-CORE].
sid	`"sid": "f5ab396c-1490-4042-b073-ae8e003c7258"`	Session ID - String identifier for a GOVSSO session. This represents a session of a User Agent. Different sid values are used to identify distinct sessions at GOVSSO.

ID Token signature validation

- Verifying the signature
  - RS256 signature algorithm is currently used
  - For the signature verification the GOVSSO public signature key must be used.
  - The used algorithm and public signature key is published at the public signature key endpoint /.well-known/jwks

ID Token signature validation

1. Read the kid value from the JWT header.
2. Validate the signature using the key corresponding to kid value from the JWT header.

ID Token validation

- Verifying the issuer of token
  - The iss value of the ID Token element must match the issuer value in /.well-known/openid-configuration
  - In DEMO this value should be https://govsso-demo.ria.ee/
- Verifying the addressee of the tokens
  - The client application must verify whether the token received was issued for them. For this purpose, it must be made sure that the aud value of the ID Token element matches the client_id issued upon registration of the client application.

ID Token validation

- Verifying the validity of the tokens
  - The verification is done using iat and exp elements in the ID Token. The client application uses its own clock to verify the validity. The following details should be verified:
    - that token issuing time has been reached:
      - iat <= (current time + permitted difference between clocks)
    - that the “expired” time has not been reached:
      - exp > (current time - permitted difference between clocks)
  - The application must choose the permitted difference between clock values.

ID Token acr level validation

- Acr value in ID token must be on equal or higher level than requested
- In case it was not specified in acr_values parameter at request, it must be equal to value high

ID Token nonce validation

- Nonce value must be the same that was used in the request

If ID Token validation is successful

- ID Token must be stored for session update flow
- ID Token information can be used for authentication

GOVSSO flow: Session update request

- Session update should be done 2 minutes before the id_token expiration
- Session update request should be done in background (with JavaScript)

GOVSSO flow: Session update request

```
GET https://govsso.ria.ee/oauth2/auth?

redirect_uri=https%3A%2F%2client.example.com%2Fcallback&
scope=openid&
state=hkMVY7vjuN7xyLl5&
response_type=code&
client_id=58e7ba35aab5b4f1671a&
ui_locales=en&
nonce=fsdsfwrerhtry3qeewq&
acr_values=substantial&
prompt=none&
id_token_hint=eyJhbGciOiJSUzI1NiIsImtpZCI6InB1YmxpYzo...TvE
```

GOVSSO flow: Session update request

URL element	*compulsory*	*example*	*explanation*
protocol, host, port and path	yes	`https://govsso.ria.ee/oauth2/auth`	`/oauth2/auth` is the OpenID Connect-based authentication endpoint of the GOVSSO service (the concept of 'authorization' originates from the OAuth 2.0 standard protocol). The URL is provided from OIDC server public discovery service: `https://govsso.ria.ee/.well-known/openid-configuration` as `authorization_endpoint` parameter.
client_id	yes	`client_id=58e7ba35aab5b4f1671a`	Client identifier. The client identifier is issued by RIA upon registration of the client application as a user of the authentication service.
redirect_uri	yes	`redirect_uri=https%3A%2F%2Fclient.example.com%2Fcallback`	Redirect URL ([OAUTH] "3.1.2. Redirection Endpoint"). The redirect URL is selected by the institution. The redirect URL may include the query component. URL encoding should be used, if necessary [URLENC]. It is not permitted to use the URI fragment component (`#` and the following component; [URI] "3.5. Fragment"). The URL protocol, host, port and path must match one of the pre-registered redirect URLs of given client application registration metadata (see `client_id` parameter).

GOVSSO flow: Session update request

scope	yes	`scope=openid`	The authentication scope. Space delimited list of requested scopes.
			
			`openid` scope is compulsory to signal that this is an OIDC authentication request.
			In the default `scope` of openid GOVSSO will issue ID Tokens with the following attributes:
			
			`sub` (physical person identifier)
			`given_name`
			`family_name`
			`birthdate`
			
			Presence of given attribute values will depend on the amount of information that is returned within TARA ID Tokens.

state	yes	`state=hkMVY7vjuN7xyL15`	Security code against false request attacks (cross-site request forgery, CSRF). Read more about formation and verification of state under “7.3 Protection against false request attacks”.
response_type	yes	`response_type=code`	Determines the manner of communication of the authentication result to the server. Only value `code` is allowed as only authorization code flow is supported by GOVSSO
ui_locales	no	`ui_locales=et`	Selection of the user interface language. The following languages are supported: `et`, `en`, `ru`. By default, the user interface is in Estonian language. The client can select the desired language. This will also set the GUI language for TARA service views.

GOVSSO flow: Session update request

acr_values	no	`acr_values=substantial`	The minimum required level of authentication based on the eIDAS level of assurance (LoA). Allowed values are: `low`, `substantial`, `high`. `high` is used by default if the value has not been set in request. GOVSSO will store the authentication level of assurance in the SSO session object as an Authentication Context Class Reference (`acr`) claim, from TARA ID Token response. Upon each GOVSSO authentication request, GOVSSO will check that the requested level of assurance (`acr_values` parameter value) is lower or equal to the `acr` claim value of the GOVSSO session. If the SSO session `acr` value (level of assurance) is lower than requested, the previous GOVSSO session is automatically terminated and a new authentication is requested from TARA. After successful authentication a new SSO session is created.
prompt	yes	`prompt=none`	Signals GOVSSO server that it MUST NOT display any authentication or consent view to the user. An error is returned if user is not already authenticated in GOVSSO or the client application does not have pre-configured consent for the requested scope, acr_values or does not fulfill other conditions for processing the request. The error code will typically be login_required, interaction_required, or another code defined in OIDC standard ([OIDC-CORE] "3.1.2.6. Authentication Error Response").
id_token_hint	yes	`id_token_hint=eyJhbGciOiJSUzI1NiIsImtpZCI6InB1YmxpYzo...TvE`	ID Token previously issued by GOVSSO being passed as a hint about the user's current or past authenticated session with the client application. If the user identified by the ID Token is logged in or is logged in by the request, then GOVSSO returns a positive response; otherwise, it WILL return an error. **Encryption of the id_token_hint parameter is not supported in GOVSSO.**

GOVSSO flow: Session update response

- Sent request

```
GET https://govsso.ria.ee/oauth2/auth?

redirect_uri=https%3A%2F%2client.example.com%2Fcallback&
scope=openid&
state=hkMVY7vjuN7xyL15&
response_type=code&
client_id=58e7ba35aab5b4f1671a&
ui_locales=en&
nonce=fsdsfwrerhtry3qeewq&
acr_values=substantial&
prompt=none&
id_token_hint=eyJhbGciOiJSUzI1NiIsImtpZCI6InB1YmxpYzo...TvE
```

- „Response“ is returned through HTTP 302 redirect
- Received response

```
GET https://client.example.com/callback?

code=71ed5797c3d957817d31&
state=hkMVY7vjuN7xyL15
```

GOVSSO flow: Session update response

URL element	*example*	*explanation*
protocol, host, port and path	`https://client.example.com/callback`	Matches the `redirect_uri` value sent in the authentication request.
code	`code=71ed579...`	The authorization code to request the ID Token.
state	`state=hkMVY7vjuN7xyL15`	Security code against false request attacks. The security code received in the authentication request is mirrored back. Read more about forming and verifying `state` from 'Protection against false request attacks'.

GOVSSO flow: Session update error response

```
GET https://client.example.com/callback?

error=authentication_required&
error_description=Authenticated+subject+does+not+match+provided+id_token_hint&
state=hkMVY7vjuN7xyL15
```

- If error code is received session should be terminated on client side immediately

ID Token request and validation

- New ID Token must be requested as in authentication
- Same validation rules must be applied before updating the session status on client side
- New ID Token must be stored to use in next session update

GOVSSO flow: Logout request

```
GET https://govsso.ria.ee/oauth2/sessions/logout?


id_token_hint=eyJhbGciOiJSUzI1NiIsImtpZCI6InB1YmxpYzo3Njc2MG...VkDzh0LYvs
post_logout_redirect_uri=https%3A%2F%2Fclient.example.com%2Fcallback&
state=0dHJpYnV0ZXMiOnsiZGF0ZV9vZl9iaXJ&
ui_locales=et
```

GOVSSO flow: Logout request

URL element	*compulsory*	*example*	*explanation*
protocol, host, port and path	yes	`https://govsso.ria.ee/oauth2/sessions/logout`	`/oauth2/auth` is the OpenID Connect-based logout endpoint of the GOVSSO service. Described in OIDC session management specification [OIDC-SESSION] "2.1. OpenID Provider Discovery Metadata" The URL is provided from OIDC server public discovery service: `https://govsso.ria.ee/.well-known/openid-configuration` `end_session_endpoint` parameter.
post_logout_redirect_uri	yes	`redirect_uri=https%3A%2F%2Fclient.example.com%2Fcallback`	Redirect URL. The redirect URL is selected by the institution. The redirect URL may include the query component. URL encoding should be used, if necessary [https://en.wikipedia.org/wiki/Percent-encoding. It is not permitted ([OAUTH] "3.1.2. Redirection Endpoint") to use the URI fragment component (`#` and the following component; [URI] "3.5. Fragment". The URL protocol, host, port and path must match one of the pre-registered redirect URLs of given client application. Client application is determined by the contents of the ID Token (token audience must belong to a registered GOVSSO client application). Different from OIDC session management specification, this parameter is considered mandatory in GOVSSO. In GOVSSO user logout flow we expect that the user is always redirected back to the client application that initiated the logout process. The `post_logout_redirect_uri` should point to the client application front page or a client application internal redirect url.

GOVSSO flow: Logout request

id_token_hint	yes	`id_token_hint=eyJhbGciOiJSU...TvE`	ID Token previously issued by GOVSSO being passed as a hint about the user's current or past authenticated session with the client application. If the user identified by the ID Token is not logged in or is logged in by the request, then GOVSSO returns a positive response; otherwise, it WILL return an error. **`id_token_hint` encryption is not supported.**
state	no	`state=hkMVY7vjuN7xyL15`	Security code against false request attacks (cross-site request forgery, CSRF). Length of `state` parameter must be minimally 8 characters. Read more about formation and verification of state under 'Protection against false request attacks'. If included in the logout request, the TARA passes this value back to the RP using the state query parameter when redirecting the user agent back to the client application. Using the state parameter is not mandatory for login callbacks. It is expected that the user was already logged out of the client application before calling GOVSSO logout endpoint.
ui_locales	no	`ui_locales=et`	Selection of the user interface language. The following languages are supported: `et`, `en`, `ru`. By default, the user interface is in Estonian language. If the user was logged into a single client application, then no GUI prompt will be displayed to the user.

GOVSSO flow: Logout response

- Sent request

```
GET https://govsso.ria.ee/oauth2/sessions/logout?


id_token_hint=eyJhbGciOiJSUzI1NiIsImtpZCI6InB1YmxpYzo3Njc2MG...VkDzh0LYvs
post_logout_redirect_uri=https%3A%2F%2Fclient.example.com%2Fcallback&
state=0dHJpYnV0ZXMiOnsiZGF0ZV9vZl9iaXJ&
ui_locales=et
```

- „Response“ is returned through HTTP 302 redirect
- Received response

```
GET https://client.example.com/callback?state=ł0dHJpYnV0ZXMiOnsiZGF0ZV9vZl9iaXJ
```

GOVSSO flow: Logout error response

- If the logout request processing is unsuccessful error will be shown to user in GOVSSO.

GOVSSO flow: Back-channel logout request

```
POST /back-channel-logout HTTP/1.1
Host: client.example.com
Content-Type: application/x-www-form-urlencoded

logout_token=eyJhbGciOiJSUzI1NiIsImtpZCI...p-wczlqgp1dg
```

GOVSSO flow: Back-channel logout request

Parameter	*compulsory*	*example*	*explanation*
protocol, host, port and path	yes	`https://client.example.com:443/back-channel-logout`	Client application must authorize access to GOVSSO to an internal URL and port. Access to the port should be limited based on IP address. The port must be protected with TLS. GOVSSO must trust the logout endpoint server certificate.
logout_token	yes	`logout_token=eyJhbGciOiJSUz...qgp1dg`	GOVSSO sends a JWT token similar to an ID Token to client applications called a Logout Token to request that they log out. Logout Token will give the client application exact information about the session (see the sid claim in ID Token) that should be logged out. The token is signed by GOVSSO with the same secret key that is used for singing issued ID Tokens.

Logout Token structure


```
{
  "aud": [
    "sso-client-1"
  ],
  "events": {
    "http://schemas.openid.net/event/backchannel-logout": {}
  },
  "iat": 1591958452,
  "iss": "https://govsso.ria.ee/",
  "jti": "c0cfc91a-cdf5-4706-ad26-847b3a3fb937",
  "sid": "9038c51d-719f-40e1-9322-a7920a2087c8"
}
```

Logout Token structure

Logout Token element (claim)	*compulsory*	*example*	*explanation*
iss	yes	`"iss": "https://govsso.ria.ee/"`	Issuer Identifier, as specified in [OIDC-CORE].
events	yes	`"events": { "http://schemas.openid.net/event/backchannel-logout": {} }`	Claim whose value is a JSON object containing the member name `http://schemas.openid.net/event/backchannel-logout.` This declares that the JWT is a Logout Token. The corresponding member value MUST be a JSON object and SHOULD be the empty JSON object `{}`. OIDC-BACK "2.4. Logout Token"

aud	yes	`"aud": [ "sso-client-1" ]`
		or
		`"aud": "sso-client-1"`
			Audience(s), as specified in [OIDC-CORE].

iat	yes	`"iat": 1591958452`	Issued at time, as specified IN [OIDC-CORE].
jti	yes	`"jti": "c0cfc91a-cdf5-4706-ad26-847b3a3fb937"`	Unique identifier for the token, as specified in [OIDC-CORE].
sid	yes	`"sid": "f5ab396c-1490-4042-b073-ae8e003c7258"`	Session ID - String identifier for a GOVSSO session. This represents a session of a User Agent. Different sid values are used to identify distinct sessions at GOVSSO.

Logout Token validation

- Signature of the Logout Token must be verified the same way as ID Token
- Issuer must be verified
- Audience must be verified
- Validity must be verified
  - The verification is done using the iat claim value of the Logout Token.
    - iat <= (current time + permitted difference between clocks)
  - The application must choose the permitted difference between clocks value.

GOVSSO flow: Logout request

- If Logout Token validation is successful client application session must be ended.
- Issued Logout Tokens are linked to ID Tokens via the sid claim. Each client application is expected to internally keep track of the ID Token sid claim and client application session relations. The client application must log out client application sessions that contain the same sid value.

GOVSSO flow: Logout response

- If client application response HTTP status code is not 200, GOVSSO will perform retry with the same Logout Token.

Testing

- For authentication:
  - TEST ID-cards or LIVE ID-cards with uploaded authentication certificates in DEMO OCSP service can be used. To upload the certificate, follow instructions here: https://demo.sk.ee/upload_cert/ .
  - Smart-ID automatic test numbers: https://github.com/SK-EID/smart-id-documentation/wiki/Environment-technical-parameters#test-accounts-for-automated-testing or personal DEMO account: https://github.com/SK-EID/smart-id-documentation/wiki/Smart-ID-demo#getting-started can be used.

Testing

- For authentication:
  - Mobile-ID automatic test numbers: https://github.com/SK-EID/MID/wiki/Test-number-for-automated-testing-in-DEMO can be used.
  - For eIDAS Czech Republic or Sweden can be used
    - Follow instructions on their sites for authentication

Testing

- Ensure that all validation rules for tokens are in place
- Session termination on timeout, back-channel logout
- User cancellation on authentication
- Errors on authentication, session update, logout

Testing and debugging tools

- Browser developer tools
- https://jwt.io/

List of endpoints for DEMO environment

Link	Comment
https://govsso-demo.ria.ee/	GOVSSO endpoint for DEMO service (OIDC issuer URL)
https://govsso.ria.ee/	GOVSSO endpoint for PROD service (OIDC issuer URL)
/.well-known/openid-configuration	Public endpoint for GOVSSO server OpenID Connect configuration information.
/.well-known/jwks.json	JSON Web Key Set document for GOVSSO service. Publishes at minimum the public key that client applications must use to validate ID Token and Logout Token signatures
/oauth2/auth	OAuth 2.0 authorization endpoint. Used for GOVSSO session update requests and authentication requests.
/oauth2/token	GOVSSO endpoint to obtain ID Token
/oauth2/sessions/logout	GOVSSO client application initiated logout endpoint

Additional information

- https://e-gov.github.io/GOVSSO/

Questions and answers

- Questions?

REPUBLIC OF ESTONIA
INFORMATION SYSTEM AUTHORITY

Thank you!