VOTING Wahlvorschlag

Authentication Model

Author	Abraxas Informatik AG
Classification	public
Version	1.1
Date	September 22, 2025

Für die digitale Schweiz. Mit Sicherheit.

Contents

1. IAM System

The SECURE Connect developed by Abraxas is used as the trusted IAM solution. This includes user management, role management and the use of SECURE Connect as an identity provider. The technical setup of applications, roles and clients is managed by SECURE Connect administrators. Initial creation of new tenants and users for the superior authority is managed by VOTING administrators while managing tenants for political parties and user assignment is delegated to the superior authority and managed in-application. It is not possible for such tenant administrators to authorize users for other tenants (or applications or roles of other tenants). Only SECURE Connect administrators can perform this kind of advanced privileged actions.
SECURE Connect does fully support the OpenID Connect (OIDC) standard for user and service authentication.

2. Procedure

2.1 Login Flow

The user login flow in the frontend uses the authorization code flow with PKCE (Proof Key for Code Exchange - https://tools.ietf.org/html/rfc7636). After successful authentication, the frontend application (user agent) is provided with an access token - JSON Web Token (JWT) - that can be used to authenticate against the backend service. Among with the access token the user receives further identity and authorization information from the identity and permission services used by the frontend for authorization and visual representation purposes.

2.2 Logout Flow

After a successful logout, all tokens and other user specific information are deleted from the client's user agent (frontend) and revoked on the identity provider server-side. When the user logs out, their active session cookies, access tokens, and refresh tokens are automatically invalidated, enhancing security by preventing unauthorized access to their account.

2.3 Refresh Flow

The frontend uses the token refresh flow to prevent long-lived access tokens. Refresh tokens are valid for 30 days and revoked at any time the user performs certain operations such as logout, password change and other session- or security related operations.

2.4 Access Token Validation

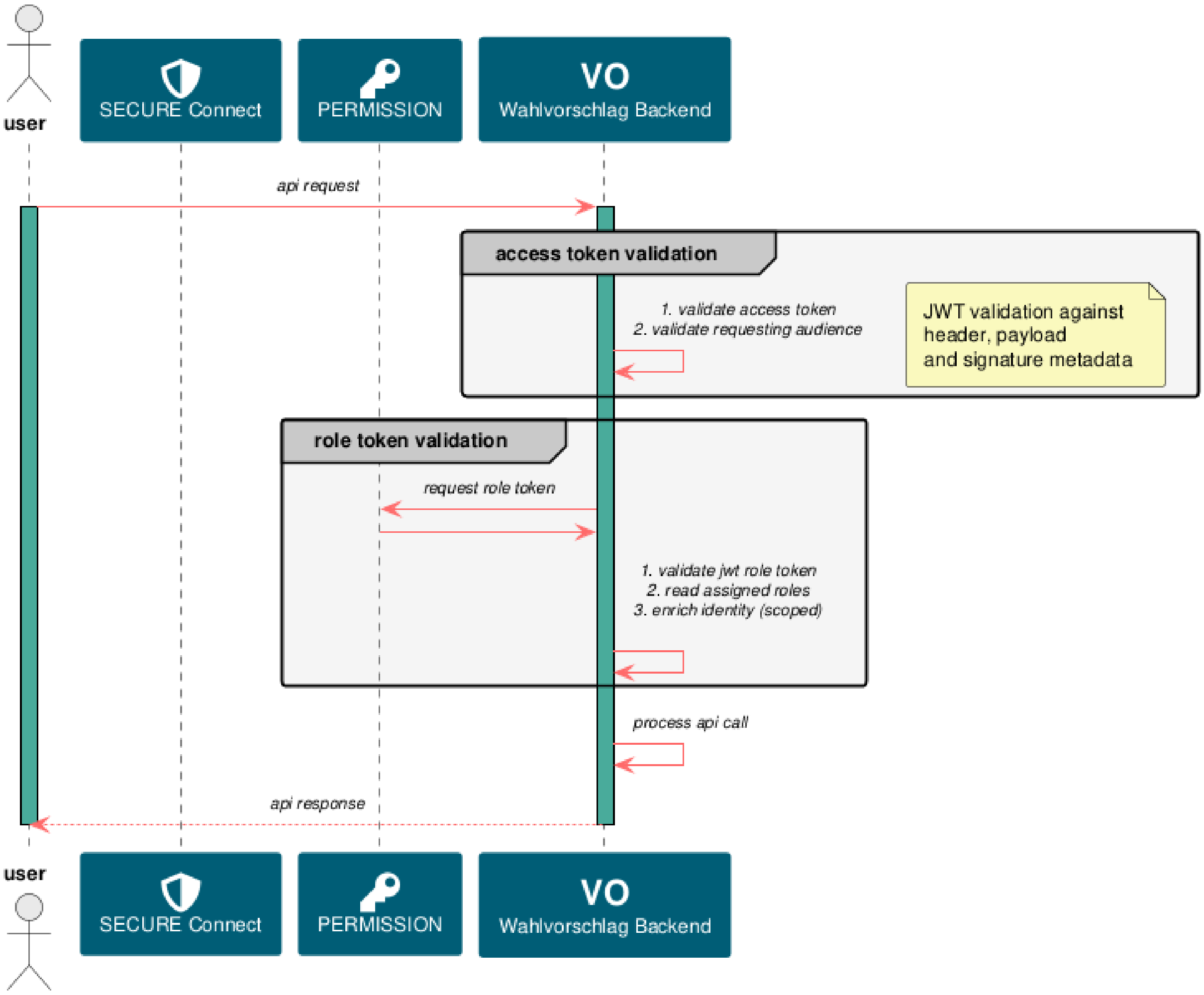
The VOTING Wahlvorschlag backend requires the following information as an HTTP header to be able to fully authenticate a user:

HTTP Header	Description	Example value
Authorization	Access token of the user	Bearer eyJ<*__TRUNCATED__*>
x-app	Application(s) to be accessed	EAWV
x-tenant	Tenant, in which context the API call should be executed. The x-tenant header corresponds to the selected tenant within the web application.	549462173064135111

The steps to successful authentication are as follows:

1. The user's access token is checked for validity (not expired, valid audience, etc.).
2. The requested application scopes are verified to see if they are included in the configured whitelist of audiences.
3. The access token and the application(s) are sent to the token endpoint.
4. In response, a role token is returned containing the user's roles for the listed applications.
5. The role token is checked for validity.
6. The roles are extracted from the token and matched with the corresponding application as defined by x-app (roles in the role token are only filtered by application, not by tenant).
7. If all validation succeeded, the user is considered authenticated.
8. Users, tenants and roles are saved for the current request to prove valid authentication against further evaluation logic.

2.4.1 Validation Flow



2.5 Authentication

IAM Instance	Application Scope	Authentication Flow	Client App	Requesting Application Audience(s)	Use Case(s)
SECURE Connect	VO Wahlvorschlag Service	Client Credential	EAWV	Identity Service Permission Service Notification Service	• Query permission and identity information to authenticate and authorize incoming requests • Authenticate for identity and permission endpoints for in-app master data

IAM Instance	Application Scope	Authentication Flow	Client App	Requesting Application Audience(s)	Use Case(s)
					management in context of superior authority members - Create new political party users - Delete existing political party users - Request new political party tenants - Change user's access for political party tenants - Send messages through the notification service
VOTING	VO Wahlvorschlag WebApp	Code with PKCE	EAWV	Identity Service Permission Service EAWV	- Query permission and identity information for visualizing user context information - Call protected backend services through REST service endpoints