

Delegated authorisation framework

Concept paper

Version: 1.1
Date: 2026-04-30
Author: Dr. Fabian Ohler / VDV ETS

Table of Contents

1	Introduction.....	4
1.1	Purpose	4
1.2	Scope	4
1.3	Definitions	4
2	Technical Overview	5
3	ESH use cases	5
3.1	Request onboarding of an Authorisation Server	5
3.2	Approve onboarding of an Authorisation Server	6
3.3	Manage an Authorisation Server	6
3.4	Provide a list of trusted Authorisation Servers.....	7
3.5	Request onboarding of a Resource Server	7
3.6	Approve onboarding of a Resource Server	7
3.7	Provide a list of Resource Servers.....	7
4	Resource Server use cases	7
4.1	Updating the list of trusted Authorisation Servers	7
4.2	Manage access to a Resource	8
4.3	Access a Resource.....	8
5	Technical notes	9
5.1	Authorisation Servers and their Scopes of Authority	9
5.1.1	Multiple Scopes of Authority per Authorisation Server	9
5.1.2	Multiple Authorisation Servers per Scopes of Authority	9
5.2	Latency.....	9
5.2.1	TLS sessions	10
5.2.2	HTTP keepalive	10
5.2.3	DPoP nonce prefetch.....	10

5.3	Nonces	10
6	Appendix: List of References.....	10

1 Introduction

This section provides a short introduction into the subject matter.

1.1 Purpose

This document introduces an authentication and authorisation framework intended to be used in the context of ((eTicket Deutschland. The framework shall allow decentralised clients to be authenticated in a delegated manner for (mostly central) resource servers to rely on.

1.2 Scope

The scope of this document is to explain the overall architecture of an authentication and authorisation framework for use in ((eTicket Deutschland. The details of the presented parts are to be provided in more detail elsewhere.

1.3 Definitions

Term	Definition
ESH	The ((eTicket Security Hub (ESH) is one of the central systems of ((eTicket Deutschland, used both internally by VDV ETS as well as by people concerned with contractual and security matters in the context of ((eTicket Deutschland.
Organisation IDs and etiCORE roles	Companies get unique organisation IDs (or Org IDs for short) as they take part in ((eTicket Deutschland. While most companies have only a single organisation ID, some companies also have more than one organisation ID. An organisation (identified by its ID) may play one or multiple ((etiCORE roles – a set of rights and obligations aligned with the structures found in the public transport in Germany.
Security Manager	A Security Manager is authorised by signatory authority to act on behalf of a company (more specifically, an organisation ID) in all matters related to the security operations in ((eTicket Deutschland.
Client	One of the OAuth 2.0 [RFC6749] roles, see there. In this context, the focus is on terminals playing the client role.
Authorisation Server	One of the OAuth 2.0 [RFC6749] roles, see there. In this context, the authorisation server logically may be seen as a part of a back-office system of a participant of ((eTicket Deutschland.
Resource Server	One of the OAuth 2.0 [RFC6749] roles, see there. Potential examples for resource servers in this context are: <ul style="list-style-type: none">- The Hotlist Service System (for its online endpoint)- The Ordered Action Management service of the Product Owner System (for its online endpoint)- An ABT/SBT ticket store- A SAM server

	- A Trusted Service Provider
Resource Owner	One of the OAuth 2.0 [RFC6749] roles, see there. Depending on the kind of resource server, different parties may play this role.
Scope of Authority	A Scope of Authority defines the boundaries within which an Authorisation Server may grant access to resources. It consists of <ul style="list-style-type: none"> - A security environment (devtest, staging, production) - An organisation ID - One or more etiCORE roles - One or more resource kinds (see Resource Server examples) per etiCORE role

2 Technical Overview

The general approach is based on the OAuth 2.0 authorisation framework and uses well established technologies. OAuth 2.0 sits in the sweet spot between stability and longevity. It will be replaced by OAuth 2.1, an evolutionary upgrade, soon. This concept is based as much as possible on the concepts of OAuth 2.0 that are not undergoing changes in OAuth 2.1 as of the current working draft.

This approach focuses on the interaction between the client and the resource server, which corresponds to the protocol steps (E) and (F) in Figure 1 of [RFC6749]. Consequently, steps (A) to (D) can be implemented by the participants of (((eTicket Deutschland tailored to their internal target architecture, e.g., incorporating hardware-backed keystores residing on the client hardware. More details can be found in section 4.

To allow for a source of trust for the resource servers towards authorisation servers, VDV-ETS is tasked with curating a list of trusted authorisation servers and their scope of authority. Furthermore, VDV-ETS provides a central service directory providing the list of valid audiences. More details can be found in section 3.

Section 5 concludes the concept with some groundwork for technical details.

3 ESH use cases

This section contains the use cases to be implemented by the (((eTicket Security Hub, the central security management system of (((eTicket Deutschland.

3.1 Request onboarding of an Authorisation Server

Summary: Request onboarding of an authorisation server for a scope of authority.

Input: Issuer URL, Scope of Authority, security concept (including TOMs)

Scenario:

1. In the ESH, the Security Manager provides an Issuer URL and a scope of authority for one of the Organisation IDs he may act for and requests onboarding of the authorisation server providing the security concept.
2. The ESH checks that the given Org ID has the given etiCORE role in the current security environment.
3. The ESH stores the request and notifies administrative staff of VDV ETS.

3.2 Approve onboarding of an Authorisation Server

Summary: Administrative staff of VDV ETS approve an authorisation server onboarding request.

Scenario:

1. The ESH retrieves the issuer discovery document [RFC8414] (issuer url + '/.well-known/openid-configuration').
2. The ESH/administrative staff of VDV ETS asserts that the issuer discovery document complies with the technical requirements (to be defined in detail as follow up work, based on [RFC8725] and [RFC9700]).
3. Administrative staff of VDV ETS audit the authorisation server and the security concept.
4. (here or beforehand) A client of the authorisation server tries to retrieve the list of resource servers (see section 3.7) to provide a prototypical access token.
5. The ESH checks that the access token complies with the requirements.
6. Administrative staff approve the authorisation server onboarding request.
7. The ESH marks the request as complete, stores the issuer and maps the issuer to the given scope of authority.

3.3 Manage an Authorisation Server

Summary: The ESH allows the Security Managers to manage their Authorisation Servers by allowing them to

- Change the Scope of Authority for an Authorisation Server and
- Offboard an Authorisation Server.

The ESH allows administrative staff of VDV ETS to manage Authorisation Servers by allowing them to

- Block / unblock an Authorisation Server (blocked servers are excluded from the list of trusted Authorisation Servers).

3.4 Provide a list of trusted Authorisation Servers

Summary: The ESH provides a list of (non-blocked) authorisation servers for use by the resource servers. This list is filtered corresponding to the related security environment. Resource Servers use a token endpoint of the ESH to authenticate.

Note: The ESH thus is a Resource Server, too, for the Resource “list of trusted Authorisation Servers”.

3.5 Request onboarding of a Resource Server

Summary: A Security Manager requests onboarding of a Resource Server for one of his organisation IDs in one of the security environments for one of the etiCORE roles associated with the organisation ID therein providing the URL of the resource server and the resource server identifier (which must be used in audience claims, must be unique within one security environment).

3.6 Approve onboarding of a Resource Server

Summary: Administrative staff of VDV-ETS approve the onboarding of a Resource Server. The Resource Server receives its credentials to authenticate during retrieval of the list of trusted authorisation servers.

3.7 Provide a list of Resource Servers

Summary: The ESH provides a list of Resource Servers to Clients including their URL, resource server identifier (audience), resource server kind and a corresponding Organisation ID and etiCORE role.

Note: The ESH thus is a Resource Server, too, for the Resource “list of Resource Servers”.

4 Resource Server use cases

This section contains the use cases to be implemented by each Resource Server.

4.1 Updating the list of trusted Authorisation Servers

Summary: Every Resource Server must periodically fetch the list of trusted Authorisation Servers provided by the ESH, which includes their scope of authority.

Scenario

1. The Resource Server retrieves the list of trusted Authorisation Servers from the ESH (using an access token from the token endpoint of the ESH).
2. For every Authorisation Server on the list with a scope of authority relevant to the Resource Server, the Resource Server
 - a. retrieves the issuer discovery document [RFC8414] (issuer url + '/.well-known/openid-configuration'),
 - b. asserts that the issuer discovery document complies with the technical requirements,
 - c. stores the issuer, jwks_uri, allowed algorithms, ...
 - d. maps the issuer to the scope of authority.
3. For every previously known Authorisation Server not considered above (either because it was not contained on the list or it was deemed irrelevant per scope of authority), the Resource Server makes sure to no longer trust them.

4.2 Manage access to a Resource

Summary: Authorised personnel of the resource server grant/revoke access to a resource for a security environment + Org ID + etiCORE role. The details of this use case depend on the resource server and are to be specified separately.

4.3 Access a Resource

Summary: Using an access token, access a resource at the Resource Server.

Input: access token, request business payload

Output: response business payload

Scenario:

1. The Terminal makes a request to access a resource at the Resource Server.
2. The Resource Server asserts that the issuer is known and active [RFC8725].
3. The Resource Server validates the token signature using the JWKS keys.
4. The Resource Server asserts that the audience matches the Resource Server **Error! Reference source not found..**
5. The Resource Server identifies the mapped the Org ID and etiCORE role for the given issuer (in the correct security environment).

6. The Resource Server asserts that all claims are valid for the identified etiCORE role (note: the authorisation server should not have issued claims invalid for its etiCORE role), following a static configuration.
7. The Resource Server handles the business request for the identified Org ID and etiCORE role.

Enforce the use of OAuth 2.0 Demonstrating Proof of Possession (RFC 9449) for the clients

- Use nonces as cryptographic nonces

5 Technical notes

This section covers some technical notes. Further details need to be worked out as follow up work.

5.1 Authorisation Servers and their Scopes of Authority

This section provides some reasoning about the allowed multiplicities between Authorisation Server and Scope of Authority.

5.1.1 Multiple Scopes of Authority per Authorisation Server

It is NOT allowed for a single Authorisation Server to be authoritative for more than one organisation ID. If we were to allow this, a Security Manager allowing an Authorisation Server to be authoritative for one of his organisation IDs cannot prevent the same server to allow foreign clients access to its resources.

It is allowed for a single Authorisation Server to be authoritative for more than one etiCORE roles under the same organisation ID.

5.1.2 Multiple Authorisation Servers per Scopes of Authority

It is allowed for multiple authorisation servers to share a scope of authority or to overlap in their scopes of authority.

5.2 Latency

For a low latency access to a resource, minimizing the number of roundtrips is critical. Therefore, the following aspects must be considered.

5.2.1 TLS sessions

Terminals must keep TLS sessions to the relevant resource servers alive as long as possible. In case a TLS session ends unexpectedly, terminals must resume the TLS session.

Resource servers must support long-lived TLS sessions and their resumption for an adequate number of terminals.

5.2.2 HTTP keepalive

The use of HTTP/1.1 is disallowed. The more current versions of HTTP (at the time of writing, HTTP/2 and HTTP/3) have their own keep alive mechanisms. Resource servers must support them. Terminals must use them.

5.2.3 DPoP nonce prefetch

Terminals must prefetch DPoP nonces from the relevant resource servers.

Resource servers must support prefetching nonces from an adequate number of terminals.

5.3 Nonces

Resource servers must treat DPoP nonces as cryptographic nonces, i.e., have them used at most once.

Resource servers must add a fresh nonce to every response.

6 Appendix: List of References

- [RFC6749] The OAuth 2.0 Authorization Framework
- [RFC8414] OAuth 2.0 Authorization Server Metadata
- [RFC8725] JSON Web Token Best Current Practices
- [RFC9449] OAuth 2.0 Demonstrating Proof of Possession (DPoP)
- [RFC9700] Best Current Practice for OAuth 2.0 Security