

Request an access token

To query another care organization's API, you need an access token from that organization's authorization server. Your Nuts node requests it for you: it shows the authorization server the credentials the use case requires.

Before you start

- You found the other care organization and its `authServerURL` (see "Find care organizations and their endpoints").
- Your care organization's wallet has the credentials the use case's policy requires (see "Load the credential into the wallet").
- You know the scope. The use case sets it.

360-graden-clientbeeld

- The wallet needs the `X509Credential`.
- The scope is `360-graden`.
- Each request needs a `NutsEmployeeCredential` for the logged-in user (see below).

Request the token

```
POST <internal Nuts interface>/internal/auth/v2/<subject ID>/request-service-access-token
```

```
Content-Type: application/json
```

```
{
  "authorization_server": "<authServerURL of the other care organization>",
  "scope": "<scope>",
  "credentials": [ <credentials that aren't in the wallet, if the use case requires them> ]
}
```

Field	Value
<code><subject ID></code> (in the URL)	The subject ID of your care organization: the one making the request, e.g. <code>careorg-a-acc</code>
<code>authorization_server</code>	The other care organization's <code>authServerURL</code> , from the Discovery Service search result
<code>scope</code>	Set by the use case

Field	Value
credentials	Optional. Credentials to present that aren't in the wallet, typically about the user. See below
token_type	Optional. DPoP (default) or Bearer. Use DPoP: a stolen DPoP token can't be used without your node's key

The user's identity

Some use cases require a credential about the logged-in user, e.g. a NutsEmployeeCredential. Pass it in `credentials`. You don't sign it, and you leave out `issuer` and `credentialSubject.id`: the node fills them in with your care organization's DID. The node's signature on the whole request protects it.

360-graden-clientbeeld

A NutsEmployeeCredential is required, for the other party's audit log:

Field	Value
name	The user's name
roleName	The user's role, e.g. "Verpleegkundige"
identifier	The user's ID in your system

```
POST <internal Nuts interface>/internal/auth/v2/careorg-a-acc/request-service-access-token
Content-Type: application/json

{
  "authorization_server": "https://nuts.other.example/oauth2/careorg-b-acc",
  "scope": "360-graden",
  "credentials": [
    {
      "@context": [
        "https://www.w3.org/2018/credentials/v1",
        "https://nuts.nl/credentials/v1"
      ],
      "type": ["VerifiableCredential", "NutsEmployeeCredential"],
      "credentialSubject": {
        "name": "J. de Vries",
        "roleName": "Verpleegkundige",
```

```
        "identifier": "123456"
      }
    }
  ]
}
```

The response

```
{
  "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCI6Ikp...",
  "token_type": "DPoP",
  "dpop_kid": "did:web:nuts.example.com:iam:657f064a-ebef-4f0f-aa87-88ed32db3142#<key id>",
  "scope": "<scope>",
  "expires_in": 900
}
```

- Use `access_token` in the `Authorization` header when you call the other party's FHIR API (or other data API).
- For a DPoP token, you also need a DPoP header for every request, signed with the key in `dpop_kid` (see "Make the FHIR call").
- The node caches tokens: calling this API again returns the same token while it's valid, so you don't have to cache it yourself.

Errors

Response	Meaning
400 Bad Request	A parameter is wrong, or the authorization server returned an OAuth error
412 Precondition Failed	Your wallet doesn't have the credentials the policy requires
502 Bad Gateway	The other party's authorization server returned an error
503 Service Unavailable	The other party's authorization server can't be reached

API reference: [Auth \(v2\)](#)

Revision #19

Created 2024-09-10 09:32:54 UTC by Rein Krul

Updated 2026-10-27 15:27:40 UTC by Rein Krul