

Description of X-Road services offered to Pääsuke

Pääsuke API version 1.0.1

- 1. Introduction
 - 1.2 Preface
 - 1.2.1 Use cases of Pääsuke that this document is covering
 - 1.2.2 Other use cases that are also resolved by Pääsuke but not covered by this document
 - 1.2.2.1 External system can use Pääsuke as a mandate system instead of implementing its own logic
 - 1.2.2.2 External system can use Tara GovSSO as authorization provider
 - 1.2.3 Principles to follow when implementing the queries
 - 1.2.3.1 Do not send null values
 - 1.2.3.2 Pääsuke is not working with historical data
 - 1.2.3.3 Each e-Service can decide its own level of support
 - 1.3 Terminology
 - 1.4 Notes
 - 1.4.1 Prototype
 - 1.4.2 OpenAPI definitions
 - 1.4.3 Mock service
 - 1.4.4 Running the mock service locally
 - 1.4.5 Playing with the mock service over X-Road
 - 1.4.6 Exporting the OpenApi definitions from SwaggerHUB
- 2. Standard X-Road services that are consumed by Pääsuke
 - 2.1. Clarifications
 - 2.1.1 Historical data is not returned
 - 2.1.2 X-road headers
 - 2.1.3 Who performs access rights check
 - 2.2 Query "getRoles"
 - 2.2.1 Why query "getRoles" is needed
 - 2.2.2 Limiting the data that needs to be transferred with each request using 'If-Modified-Since' header
 - 2.2.3 Hidden roles
 - 2.3 Query "getMandatesByRepresentee"
 - 2.3.1 View "Ettevõtte esindajad ja volitatud isikud"
 - 2.3.3 View "Minu esindajad"
 - 2.3.4 View "Ettevõtte <ettevõteteniimi> poolt antud volituste edasivolitused"
 - 2.3.5 Response structure of query getMandatesByRepresentee
 - 2.3.6 Rendering links for deleting a mandate
 - 2.4 Query getMandatesByDelegate
 - 2.4.1 View "Ettevõttele antud volitused"
 - 2.4.2 View "Mulle antud volitused"
 - 2.4.3 View "Isikule <füüsiline või juriidiline isik> antud edasivolitused"
 - 2.4.4 Response structure of query getMandatesByDelegate
 - 2.4.5 Differences between waiving and withdrawing a mandate
 - 2.4.6 Links for waiving a mandate ("Loobu")
 - 2.4.7 Links for adding a sub-delegate ("Volita edasi")
 - 2.5 Query editMandate
 - 2.5.1 Path parameters
 - 2.5.2 Payload
 - 2.5.3 View to either withdraw mandates from the delegate or for the delegate to waive the mandates
 - 2.5.4 Deleting a mandate that has been further sub-delegated
 - 2.5.5 Deleting a mandate that has not been further sub-delegated
 - 2.5.6 Digitally signing the request
 - 2.6 Query addMandate
 - 2.6.1 View to add mandates to a delegate
 - 2.6.2 Path parameters
 - 2.6.3 Payload
 - 2.7 Query addSubDelegate
 - 2.7.1 View where adding a sub-delegate can be started
 - 2.7.2 Digitally signing the request
 - 2.7.3 Path parameters
 - 2.8 Listing mandates that have been created using adding a sub-delegate (Edasivolitused)
 - 2.8.1 View "Ettevõtte <ettevõteteniimi> poolt antud volituste edasivolitused"
 - 2.8.2 View "Isikule <füüsiline või juriidiline isik> antud edasivolitused"
 - 2.9 Support for TÖR (Töötamise Register) use-case
- 3. Types
 - 3.1 Person
 - 3.1.1 Person identifier
 - 3.2 Namespace
 - 3.3 Role and RoleDefinition
 - 3.3.1 Role and role code
 - 3.3.2 RoleDefinition parameters
 - 3.3.3 Representation rights loaded from Business Registry
 - 3.4 MandateTriplet
 - 3.5 Mandate
 - 3.6 MandateLinks

- 3.6.1 delete
 - 3.6.2 addSubDelegate
 - 3.6.3 update
 - 3.6.4 origin
 - 3.7 Translation
 - 3.8 Authorization
 - 3.8.1 Person has a role that allows creating, sub-delegating, and removing mandates
 - 3.8.2 Person is on the board and has the right to represent the legal entity alone
 - 3.8.3 Several people who are on the board and they have partial rights (ühisesindusõigus) but together they can represent the legal entity
 - 3.9 ValidityPeriod
 - 3.10 Problem
 - 3.11 Document
- 4. Details about roles and namespaces
 - 4.1 Namespaces with parent-child relation
 - 4.2 Special parent-child namespaces for Admin Portal access rights
 - 4.3 Dedicated namespaces
 - 4.3.1. Dedicated namespace for representation rights loaded from Business Registry (Äiregistrist)
 - 4.3.2. Dedicated namespace for mandates managers
 - 4.3.3. Dedicated namespace for mandates managers
 - 4.3.4. Dedicated namespace for representation rights between natural persons

1. Introduction

This document is accompanied by OpenAPI definitions:

<https://app.swaggerhub.com/apis/aasaru/x-road-services-consumed-by-paasuke/<version number>>

and these definitions refer to a common domain definition:

<https://app.swaggerhub.com/domains/aasaru/paasuke-common-types/<version number excluding the minor version>> 1.1 Versions

Version	Date	Description and changes
0.2.0		First public draft with the services that are offered by parties who keep mandates on their side and want to publish that info to Pääsuke
0.2.1		Added chapter "3. X-road services offered by Pääsukese to query mandates that are stored in Pääsuke"
0.2.2		Added chapters "4. X-road services to modify mandates that are stored in Pääsuke" and "5. Integration with Pääsuke without implementing any X-road services"
0.3.1		Improved terminology and introduced different types of namespaces (parent, child, standalone and external). Header parameters changed back to query parameters (except "If-Modified-Since") Added chapter 3. "X-road services offered by Pääsukese to query mandates that are stored in Pääsuke" together with new endpoint getNamespaces Person type INDIVIDUAL changed to NATURAL_PERSON. Smaller adjustments and parameters.
0.4.0		Shortened "namespace" to "ns" everywhere. Added ns path parameter to all methods that change state. New endpoints addMandateToDelegate, addMandateSubdelegate, removeAllMandatesFromDelegate Added chapter 1.2. Preface to this document.
0.4.1		Changed the Translations object ("2letterLangCode":"translation") changed parent_namespace parentNamespace Person type LEGAL_ENTITY changed to LEGAL_PERSON (so it matches better with NATURAL_PERSON)
0.5.0	19.01. 2023	Person IdentityCode changed into person identifier. Added description of responses to queries. Added description of custom data types used inside the API Added description of how sub-delegating a mandate takes place.
0.5.1	20.01. 2023	validityPeriod, editMandate, Authorization
0.5.2	24.01. 2023	Added mandatevalidityPeriodLimit

0.6.0	31.01.2023	<p>Changed Person.identifier - now using URI-s instead of "internal:", "email:" prefixes.</p> <p>RoleMetaData moved into RoleDefinition and removed state field from it.</p> <p>Added roleDefiniton fields Role.deletableBy, Role.representeeType (this has an additional enum value GOVERNMENT_PERSON).</p> <p>The output of "/roles" query changed so that it returns an array of the following: namespace: "{namespaceCode}", roles: <array of all the roles in namespace></p> <p>Added description Problem data type that is returned in case of any errors.</p> <p>Removed support for editing a mandate through xRoadPutEdit link.</p> <p>Removed validityPeriodLimit.</p>
0.6.1	08.02.2023	<p>Role codes now always start with namespace + ":"</p> <p>Role codes can contain any UTF-8 characters (including spaces).</p> <p>Added role parameter "deletableByDelegate".</p> <p>Added roles that are checked when Pääsuke is used through MISP2 portal.</p>
0.7.0	14.02.2023	<p>Role codes now always start with the namespace code followed by a colon.</p> <p>Changes to links:</p> <ul style="list-style-type: none"> • xRoadDeleteMandate delete • xRoadPutEdit update • xRoadPostSubDelegate addSubDelegate • uiExternalView origin • uiExternalEdit - removed <p>The internal names of the methods were changed:</p> <ul style="list-style-type: none"> • "removeMandate" was renamed to deleteMandate. • "addMandateSubDelegate" "addSubDelegate" <p>Added new Person type UNKNOWN.</p> <p>Removed pagination (attributes "limit" and "skip" for GET queries).</p> <p>Changed URL path "/ns/" (namespace) to "/nss/" (namespaces)</p> <p>Added role definition attribute "deletableByDelegate".</p>
0.7.1	16.02.2023	a separate x-road request is made for adding each mandate (rather than grouping several mandates into a single request)
0.7.2	17.02.2023	<p>Removed "update" link that was meant for updating the record. It might be added back in future versions of API.</p> <p>Added Person.type OTHER as a value.</p> <p>Added RoleDefinition.visible</p>
0.7.3	21.02.2023	Common types (including request and response types) moved to a different domain definition. Improved the documentation within the OpenAPI definitions.
0.7.4	8.03.2023	<p>To RoleDefinition added fields canAssignIfHasRoleAndOneOf, canDeletelfHasRoleAndOneOf.</p> <p>Added the global restriction that the sub-delegate can only be a natural person and never a legal person.</p> <p>The mandate can include a field "subDelegatorIdentifier" to indicate the identifier of the person who added this relation via sub-delegation.</p>
0.7.5	6.04.2023	Set 100 to be the maximum number of Mandate objects in a single MandateTriplet (chapter 3.4).
0.7.6	19.04.2023	<p>Add new filter parameter "delegate" to getRepresenteeDelegatesWithMandates (getMandatesByRepresentee since 1.0.0).</p> <p>To better facilitate eesti.ee RR partner use-case the role definition parameters canAssignIfHasRoleAndOneOf and canDeletelfHasRoleAndOneOf were removed and replaced with new parameters assignableOnlyIfRepresenteeHasRoleIn and delegateCanEqualToRepresentee (see chapter 3.3.2)</p>
0.8.0	8.05.2023	<p>Added new parameter document to payloads of addMandate and addSubDelegate.</p> <p>This contains information about the digitally signed asice container.</p>
0.8.1	11.05.2023	RoleDefinition.visible changed from optional to compulsory.

0.9.0	23.05.2023	<p>deleteMandate method was renamed to editMandat and changed from DELETE to PUT (that takes a request body)</p> <p>In Chapter 1.3. split deleting a mandate into two separate terms:</p> <ol style="list-style-type: none"> 1. mandate withdrawal 2. mandate waiving. <p>In Chapter 3.3.2 changed RoleDefinition:</p> <ul style="list-style-type: none"> • added <ul style="list-style-type: none"> ◦ addingMustBeSigned ◦ withdrawalMustBeSigned ◦ waivingMustBeSigned ◦ subDelegableBy (list of strings) • added new fields and deprecated some old ones: <ul style="list-style-type: none"> ◦ assignableBy (now deprecated) replaced with addableBy ◦ assignableOnlyIfRepresenteeHasRoleIn (now deprecated) replaced with addableOnlyIfRepresenteeHasRoleIn ◦ deletableBy (now deprecated) now renamed to withdrawableBy • changed <ul style="list-style-type: none"> ◦ visible:false (now deprecated) changed to hidden:true ◦ deletableByDelegate(boolean, now deprecated) changed to waivableBy (list of strings) • following RoleDefinition attributes are now compulsory <ul style="list-style-type: none"> ◦ representeeType ◦ delegateType
0.9.1	26.05.2023	Removed "/nss/{ns}" from the beginning of add a sub-delegate link and from the beginning of the edit mandate link
0.9.2	13.06.2023	<ul style="list-style-type: none"> • Added RoleDefinition.validityPeriodFromNotInFuture, RoleDefinition.validityPeriodThroughMustBeUndefined. • When mandates are queried and there are no mandates to return then an empty list should be returned (instead of returning HTTP status code 404).
0.9.3	19.06.2023	<p>RequestBody of PUT changed from EditMandate to DeleteMandate to ease implementation.</p> <p>waivableBy, withdrawableBy, subDelegableBy - these no longer default to addableBy when unset but the values must be defined for each parameter.</p> <p>Removed deprecated RoleDefinition parameters (assignableBy, assignableOnlyIfRepresenteeHasRoleIn, deletableBy, visible, deletableByDelegate).</p>
0.9.4	4.09.2023	<p>Added subDelegatingMustBeSigned to RoleDefinition.</p> <p>Support for following RoleDefinition attributes lifted from 1.1. to 2.0: canSubDelegate, subDelegateType, subDelegableBy, validityPeriodFromNotInFuture and validityPeriodThroughMustBeUndefined.</p> <p>Removed RoleDefinition attribute modified.</p>
0.9.5	22.09.2023	<p>Action "DELETE" under editMandate which is used to end the validity of the mandate has been split into two parts:</p> <ul style="list-style-type: none"> • action="DELETE_WITHDRAW" (if the mandate is being withdrawn by the representee) • action="DELETE_WAIVE" (if the mandate is being waived by the delegate) <p>See Chapter 2.5 for more info.</p>
0.9.6	20.10.2023	<p>Added representee to RoleDefinition.representeeIdentifierIn</p> <p>Remove GOVERNMENT_PERSON person type sub-type.</p>
0.9.7	11.12.2023	Chapters related to role configurations moved to a separate document "Introduction to Pääsuke and configuring roles"
0.9.8	15.12.2023	<p>Added new filter parameter "subDelegatedBy" to query getRepresenteeDelegatesWithMandates (getMandatesByRepresentee since 1.0.0).</p> <p>Added a screenshot about when and how mandate.subDelegatorIdentifier is displayed in the user-identifier.</p>
0.9.9	02.01.2024	<p>editMandate (PUT) query now returns an object together with status code 200. This object can include list of mandates that were deleted additionally. This can happen when withdrawing or waiving a mandate that was further sub-delegated. In this case the sub-delegated mandates must be deleted in addition to the original mandate.</p> <p>editMandate (PUT) query can no longer return status code 202 (Accepted).</p>
1.0.0	16.02.2024	<p>Query getRepresenteeDelegatesWithMandates renamed into getMandatesByRepresentee</p> <p>Query getDelegateRepresenteesWithMandates renamed into getMandatesByDelegate.</p> <p>Added new field "subDelegable" to a mandate.</p> <p>Added optional filter parameter "ns" to getMandatesByRepresentee and getMandatesByDelegate.</p>
1.0.1	21.03.2024	<p>Added mandateTriplet.mandate.subDelegator</p> <p>Deprecated mandateTriplet.mandate.subDelegatorIdentifier in favour of mandateTriplet.mandate.subDelegator</p>

1.2 Preface

1.2.1 Use cases of Pääsuke that this document is covering

Pääsuke displays the mandates that are stored in external e-services (at Tax and Customs Board for example). If the external e-service supports it, Pääsuke offers the following additional operations besides displaying mandates:

- deleting a mandate
- adding a new mandate
- adding a sub-delegate for a mandate

In order to resolve these use cases Pääsuke has created a standard for the queries that e-services must offer to Pääsuke over x-road.

This document is the description of that standard.

Once the external e-service has implemented some or all of the queries in the standard then Pääsuke can be configured to call these queries.

1.2.2 Other use cases that are also resolved by Pääsuke but not covered by this document

1.2.2.1 External system can use Pääsuke as a mandate system instead of implementing its own logic

e-services can use Pääsuke to store the mandates centrally in eesti.ee and use the user interface of eesti.ee to manage them (give out new or delete old ones).

Pääsuke offers x-road services to these e-services to query mandates from Pääsuke when a person tries to authorize himself in that service. Pääsuke also offers a list of persons (like management board members, procurers, etc) defined in the Business Registry as a response to this query.

1.2.2.2 External system can use Tara GovSSO as authorization provider

Currently, for most Estonian government e-services the authentication service is provided by Tara. For any e-service using Tara, there is no need to implement x-road services as Tara provides signed proof to e-services about the authenticated person.

GovSSO is Tara with SSO and it provides single sign-on functionality on top of Tara. It would be technically possible for GovSSO to offer additional UI flows for authenticated users to select a person to be represented in the upcoming session. This way GovSSO would provide the selected person as part of the OpenID connect flow together with details of the authenticated person.

If the user later wants to switch represented person to a different person then that would be possible as the e-service would anyway have to keep the session alive with GovSSO. To switch a representee the e-service would have to send the user's browser back to GovSSO for that and the user would return with details of the selected representee that would be signed by GovSSO.

This integration pattern is currently seeking interested parties. Please connect with Pääsuke team if you would be interested in using that flow.

1.2.3 Principles to follow when implementing the queries

1.2.3.1 Do not send null values

Not recommended	Recommended
<pre>"title": { "et": "Tere", "en": "Hello", "ru": null }</pre>	<pre>"title": { "et": "Tere", "en": "Hello" }</pre>

1.2.3.2 Pääsuke is not working with historical data

When implementing the queries:

- only return mandates that are currently valid or will become into effect in the future.
- do not include mandates that have been active in the past and have been deleted

1.2.3.3 Each e-Service can decide its own level of support

There are the following options for any e-service:

The required part is to offer queries for Pääsuke to query and display the mandates. This means the e-service must implement two queries for pulling the mandates and one query for Pääsuke to pull the role definitions.

Additionally, any or all of the following can be supported by the e-service

1. Deleting mandates
2. Adding new mandates

3. Adding a sub-delegate for a mandate

1.3 Terminology

- **Pääsuke** - central access rights management system hosted in eesti.ee
- **RIA** - Information System Authority (Riigi Infosüsteemi Amet), agency that develops and runs eesti.ee and Pääsuke
- **institution** - some party who has a self-service system that either queries mandates from Pääsuke and/or has mandates declared in the system and publishes them in Pääsuke.
- **representee** - a person (private or legal) who has given a mandate to a delegate to be represented by that delegate (or its sub-delegates)
- **delegate** - a person (private or legal), to whom a representee has given the mandate to represent itself. Delegate normally always has the right to represent oneself (except if the person doesn't possess active legal capacity - in Estonian "piiratud teovõime")
- **namespace** - a group of roles that are maintained by a single institution.
- **privilege** - individual right to perform some action in e-service
- **role** - a group of privileges to be used in a self-service that can be granted to delegate by the representee. Role always belongs to a namespace. More information about configuring roles can be found in document "Introduction to Pääsuke and configuring roles"
- **mandate** - a role that is given to a delegate by some representee. Mandates can have a start date and end date, and some mandates can be sub-delegated (in Estonian "*edasi delegeerima*").
- **mandate withdrawal** - if the representee (or a person representing the representee) deletes the mandate that has been given to some delegate (in Estonian "*volituse tagasivõtmine*")
- **mandate waiving** - if the delegate (or a person representing that delegate) deletes a mandate that has been given to the delegate (in Estonian "*volitusest loobumine*")

1.4 Notes

1.4.1 Prototype

Pääsuke offers its prototype publicly - it is available here: <https://paasuke.github.io/proto/>. The screenshots used in this document are taken from the prototype.

The prototype is to illustrate how different mandates would be displayed to the user and what assigning a role looks like and how it looks like to withdraw/waive a mandate or add a sub-delegate to a mandate.

1.4.2 OpenAPI definitions

OpenAPI definitions (verify that the version in the link is up to date):

- <https://app.swaggerhub.com/apis/aasaru/paasuhalduse-x-tee-teenused/<version>#/Offered%20to%20P%C3%A4%C3%A4suke>
- [https://app.swaggerhub.com/apis/aasaru/paasuhalduse-x-tee-teenused/<version>#/Offered%20to%20P%C3%A4%C3%A4suke%20\(additional\)](https://app.swaggerhub.com/apis/aasaru/paasuhalduse-x-tee-teenused/<version>#/Offered%20to%20P%C3%A4%C3%A4suke%20(additional))

1.4.3 Mock service

In order to better illustrate what kind of responses are expected from external parties implementing these queries a mock service has been developed.

It is possible to run the mock locally using Docker or Java (see [Running the mock service locally](#)).

The mock is also accessible over ee-dev X-Road (see chapter 1.4.5).

The same mock plays different parties (EMTA and STAT for example):

- If you want the mock to act like EMTA mock then set the value of "X-Road-Id" header to something that starts with "EMTA"
- If you want the mock to act like a mock for STAT subsystem ESTAT then set the value of "X-Road-Id" header to something that starts with "STAT"
- In the future, the mock will also serve other agencies like "PRIA", "MAJ" etc

1.4.4 Running the mock service locally

Instructions can be found here: <https://github.com/e-gov/PH/tree/main/ph-xroad-api-mock>

1.4.5 Playing with the mock service over X-Road

A mock service has been set up in ee-dev X-Road that mimics the expected behavior of a system providing such services.

You can send requests against that service to better understand how the service has to work.

Throughout this document, the X-Road-specific headers (headers beginning with X-Road-...) are removed from the examples of HTTP requests.

If you want to test the queries then you always need to add the following x-road headers and the accept parameter:

```
curl \
-H "accept: application/json" \
-H "X-Road-Client: ee-dev/GOV/70001234/generic-consumer" \
-H "X-Road-UserId: EE39912310123" \
-H "X-Road-Id: EMTA_08544bbd2f41473800309d16bd81c64c0f54193d84b53f8ad22aacdf5e" \
-X GET "https://security-server/r1/ee-dev/GOV/70006317/volitused-mock/volitused-estat/v1/roles"
```

You need to make the following replacements:

- ee-dev/GOV/70001234/generic-consumer - replace with your own details. RIA needs to grant access to this x-road client.
- replace <https://security-server> with your security server IP/DNS.
- You need to set X-Road-UserId to your own personal id code.

1.4.6 Exporting the OpenApi definitions from SwaggerHUB

The definitions of the API are split between two files - the common data types are defined in another URL.

This is how you can download all the definitions as a single file (this is called 'Resolved').

1. Open definitions <https://app.swaggerhub.com/apis/aasaru/x-road-services-consumed-by-paasuke/>
2. Check that the correct version is open. If needed change the version from the drop-down menu
3. Do not copy-paste the definitions (as this would end up with an unresolved copy)
4. Choose Export Download API either pick "JSON Resolved" or "YAML Resolved".

2. Standard X-Road services that are consumed by Pääsuke

These services are used to show all the mandates from a central system. This way:

- any representee has visibility all over the Estonian e-services of the mandates that are currently valid.
- any delegate has information about all the mandates assigned to him by different representees.

Pääsuke uses the following services to query systems that among other things store mandates. These services are called standard services. Although data providers are different (Statistics Estonia, Estonian Tax and Customs Board, etc) these systems have all agreed to use the same query and data format.

This data that is pulled is displayed in Pääsuke UI (that is going to reside under eesti.ee)

2.1. Clarifications

2.1.1 Historical data is not returned

The services only return mandates that are currently valid or will become valid in the future. Records that are no longer valid are not available through this API.

2.1.2 X-road headers

If an actual person is making requests in Pääsuke then Pääsuke always adds headers:

- X-Road-UserId - identifier of the user currently logged in to Pääsuke (for example EE50001029996)
- X-Road-Represented-Party - identifier of the legal person currently being represented (for example EE11065244)

However, there might be requests from Pääsuke that are made by some automatic process. Then these two headers are missing.

One example of such a request could be a situation where:

1. The Employment Register (TÖR) has identified that person P left company C one day ago
2. The Employment Register triggers a request to Pääsuke asking if P has any valid mandates under C (no information about the user as it is a background process).
3. Pääsuke makes a request to other systems that store mandates on their side (and it doesn't add these headers) to find out if P has any valid mandates under C
4. If any matches are found then TÖR sends out an e-mail to management board members of C with a warning (like: we noticed that some person recently left your company but it seems the person still has valid mandates. Please go to Pääsuke and review the mandates of your company).

2.1.3 Who performs access rights check

Pääsuke is built to verify if the person is allowed to add, edit, or withdraw/waive any mandate according to role configuration.

The party that provides x-road services is welcome to add their own validations. This forms a two-layer authorization check.

2.2 Query "getRoles"

OpenAPI definition: <https://app.swaggerhub.com/apis/aasaru/x-road-services-consumed-by-paasuke/<version>#/default/getRoles>

CURL query

```
curl -H "If-Modified-Since: 2022-11-12T00:00:00+02:00" \
-X GET \
"https://security-server/r1/ee-dev/GOV/70006317/volitused-mock/volitused-estat/v1/roles"
```

The return type is the array of [RoleDefinition](#) (described in chapter 3.3.2).

Pääsuke uses this query to periodically fetch all translations of roles and also the metadata about roles. This metadata tells Pääsuke who is allowed to assign a particular role.

2.2.1 Why query "getRoles" is needed

The queries that return mandates, these queries use role codes in the mandate payload. For the Pääsuke UI to translate these codes into different languages it uses this getRoles query to fetch the translations.

2.2.2 Limiting the data that needs to be transferred with each request using 'If-Modified-Since' header

Pääsuke does not include the 'If-Modified-Since' header in the first request.

When the service returns the list of roles, Pääsuke goes over all the returned roles and memorizes the latest modification date (if at least one of the roles had a value for metadata.modification).

If the latest modification date has been stored by Pääsuke then on the next request it includes this value on the 'If-Modified-Since' header in the request. This is done to indicate the date and time of the latest role modification Pääsuke has already copied over.

The service can:

- Respond with HTTP Status code 304 if no roles have been changed since that time. The service provider is also allowed to ignore that property and never respond with HTTP Status code 304.
- Otherwise, all results (that match the filters) are returned (even the ones that have modified time earlier than the If-Modified-Since parameter).

2.2.3 Hidden roles

There are use cases where mandates are used to send additional info that is needed by Pääsuke but not displayed to the end user. For this Pääsuke allows the roles to be configured as hidden.

Let's look at the following example. PRIA only allows new mandates to be added from Pääsuke only for (legal and natural) persons who are registered as PRIA customers.

The definition of the role "PRIA:PRIA.customer" has the following parameters:

Field name	Field value
code	PRIA:pria.customer
title.et	PRIA klient
hidden	true

And a MandateTriplet with that role would be returned as:

CURL query

```
{
  "representee": {
    "type": "LEGAL_PERSON",
    "legalName": "Agro Agro AS",
    "identifier": "EE11430169"
  },
  "delegate": {
    "type": "LEGAL_PERSON",
    "legalName": "Agro Agro AS",
    "identifier": "EE11430169"
  },
  "mandates": [
    {
      "role": "PRIA:PRIA.customer"
    }
  ]
}
```

Note that the values of the delegate and the representee are the same!

2.3 Query "getMandatesByRepresentee"

CURL query

```
curl -X GET \
  "https://security-server/r1/ee-dev/GOV/70006317/volitused-mock/volitused-estat/v1/representees/{representee}/delegates/mandates"
```

Returns all delegates (with mandates), who have a right to represent the representee currently or are scheduled to become active in the future.

NB! The query should not return mandates that have been valid in the past.

This query takes optional query parameter "ns". If this parameter is given then the returned mandates must start with one of the values in the list.

Systems (like PRIA, ANNUAL_REPORTS, EMTA) that only serve a single namespace can ignore this query parameter.

2.3.1 View "Ettevõtte esindajad ja volitatud isikud"

This query is used to serve the following view in the Pääsuke UI. It displays all mandates that the representee has given out to others or that are assigned by law (Äriregistrist).

Väikefirma OÜ (EE12345678)

Ettevõtte esindajad ja volitatud isikud

Ettevõttele antud volitused

Lisa uus volitus

Ava otsing

<div> <div></div> <div>Mari Maasikas (49028099999)</div> </div>	<div>Äriregistrist: Juhatus liige</div>	Vaata
<div> <div></div> <div>Kaupo Kuusik (37925050002)</div> </div>	<div>Äriregistrist: Prokurist</div>	Vaata
<div> <div></div> <div>Jüri Vaarikas (39211110000)</div> </div>	<div> Maksu- ja Tolliamet: Raamatupidaja pakett Statistikaamet (eSTAT): Andmeesitaja e-Äriregister: Majandusaasta aruannete sisestaja Asutusteülene: Raamatupidaja (alates 6.11.2023) </div>	Muuda Kopeeri Eemalda
<div> <div></div> <div>Kalle Kuusk-Konstantinov (37425050015)</div> </div>	<div> Statistikaamet (eSTAT): Andmeesitaja Transpordiamet: Üldvolikiri Transpordiamet: Volikiri konkreetse sõiduki kohta Riigihanked: Peakasutaja e-Äriregister: Majandusaasta aruannete sisestaja </div>	Muuda Kopeeri Eemalda
<div> <div></div> <div>Raamatupidajad OÜ (EE23456789)</div> </div>	<div> Asutusteülene: Raamatupidaja PRIA: Dokumentide vaataja </div>	Muuda Kopeeri Eemalda

2.3.3 View "Minu esindajad"

In the future natural person can use Pääsuke to see what kind of natural persons he/she has given mandates to represent himself/herself. To show the mandates the application also performs the query described at the beginning of this paragraph (2.3).

2.3.4 View "Ettevõtte <ettevõttenimi> poolt antud volituste edasivolitused"

There is one additional view that is served by the same query (and by setting query parameter "subDelegatedBy"), this use case is described in chapter 2.8.1.

It is only important if the external service is planning to allow adding sub-delegates.

2.3.5 Response structure of querygetMandatesByRepresentee

This query and the next query (described in 2.4) have identical response structures, both return a list of MandateTriplets. The MandateTriplet type is described in chapter 3.4.

This query cannot return links of type "addSubDelegate" but it can return links of type "delete".

2.3.6 Rendering links for deleting a mandate

If at least one of the mandates has a link "delete" then the UI displays a button for initiating deleting of the mandate.

2.4 Query getMandatesByDelegate

```
curl -X GET \
  "https://security-server/r1/ee-dev/GOV/70006317/volitused-mock/volitused-estat/v1/delegates/{delegate}/representees/mandates"
```

Returns all representees (with mandates) that the delegate has the right to represent.

This query serves views that are described in chapters 2.4.1 and 2.4.2.

This query takes optional query parameter "ns". If this parameter is given then the returned mandates must start with one of the values in the list.

Systems (like PRIA, ANNUAL_REPORTS, EMTA) that only serve a single namespace can ignore this query parameter.


2.4.1 View "Ettevõttele antud volitused"

A legal entity (like an accountant bureau) is looking, at what kind of mandates other legal entities have given him.

Väikefirma OÜ (EE10391131)

Ettevõtte esindajad ja volitatud isikud

Ettevõttele antud volitused

 **Pikad Puud OÜ (EE88765432)**

Statistikaamet (eSTAT): **Andmeesitaja** ⓘ (alates 13.12.2028)

[Vaata edasivolitusi](#) [Volita edasi](#) [Loobu](#)


2.4.2 View "Mulle antud volitused"

A natural person opens Pääsuke to see what kind of mandates he has been given anywhere in the Estonian e-services (that are present in Pääsuke).

Mari Maasikas (49028099999)


Mulle antud volitused

Minu esindajad

 **Raamatupidajad OÜ (EE23456789)**

Äriregistrist: **Juhatuselise liige (ühisesindus)** ⓘ
Statistikaamet (eSTAT): **Andmeesitaja** ⓘ
Asutusteülene: **Raamatupidaja** ⓘ

[Vaata edasivolitusi](#) [Volita edasi](#) [Loobu](#)

 **Väikefirma OÜ (EE12345678)**

Äriregistrist: **Juhatuselise liige** ⓘ

[Vaata](#)

2.4.3 View "Isikule <füüsiline või juriidiline isik> antud edasivolitused"

There is one additional view that is served by the same query (and by setting query parameter "subDelegatedBy"), this use case is described in chapter 2.8.2.

It is only important if the external service is planning to allow adding sub-delegates.

2.4.4 Response structure of query getMandatesByDelegate

This query and the previous query (described in 2.3) have identical response structures, both return a list of MandateTriplets. The MandateTriplet type is described in Chapter 3.4.

This query can return links of type "addSubDelegate" and "delete".

2.4.5 Differences between waiving and withdrawing a mandate

If a person has added a mandate to a delegate and then wishes to delete this mandate then this is called withdrawing (tagasivõtmine). If the delegate wants to delete this mandate that has been given to her then this is called waiving (loobumine).

2.4.6 Links for waiving a mandate ("Loobu")

If at least one of the mandates has a link "delete" then the UI displays a button that initiates giving up a mandate. It is similar to deleting a mandate but the deleting is initiated from the delegate side (the receiver of the power of the attorney).

2.4.7 Links for adding a sub-delegate ("Volita edasi")

If at least one of the mandates in the list has a link "addSubDelegate" then the UI displays a button to initiate adding a sub-delegate ("Volita edasi").

2.5 Query editMandate

Currently only deleting (withdrawing or waiving) is allowed.

2.5.1 Path parameters

```
curl -X PUT \
  "https://security-server/r1/ee-dev/GOV/70006317/volitused-mock/volitused-estat/v1/representees/{representeeId}/delegates/{delegateId}/mandates/{mandateId}"
```

Values of `representeeId`, `delegateId`, and `mandateId` are taken by Pääsuke from the output of the query that produced the list (link with `rel "delete"`). The value of `ns` is taken from the role of the mandate (the first part of the role until the first colon (which is excluded)).

2.5.2 Payload

```
{
  "action": "DELETE_WITHDRAW",
  "authorizations": [
    {
      "userIdentifier": "EE39912310123",
      "hasRole": "BR_REPRIGHT:SOLEREP"
    }
  ],
  "document": {
    "uuid": "5b72e01c-fa7f-479c-b014-cc19efe5b732",
    "singleDelegate": true
  }
}
```

This request tells the provider of the service to end the validity of the mandate.

The action is a required attribute.

- If the person who gave the mandate (representee) is ending the validity of the mandate then the action is set to 'DELETE_WITHDRAW'.
- If the mandate is being ended by the person who was the delegate of the mandate then the action is set to 'DELETE_WAIVE'.
- Until version 0.9.4 the value was always set to 'DELETE'.

★ - The property "document" is present if the deleting of the mandate was digitally signed. Signing will be added in Pääsuke version 2.0

2.5.3 View to either withdraw mandates from the delegate or for the delegate to waive the mandates

Serves the following view of the Pääsuke UI. This view allows the user to individually pick the mandates to be removed.

Esindaja volitused

×

Esindaja

Esindatav

Raamatupidajad OÜ (EE23456789)

Väikefirma OÜ (EE12345678)

Asutusteülene: Raamatupidaja ⓘ

Rolli kehtivus järgnevates asutustes:

☒ Maksu- ja Tolliamet - samad õigused, mis on raamatupidaja pakettis (alates 14.12.2022 - tähtajatu)
 ☒ Eesti.ee - töövõimeuslehtede kinnitamine (alates 14.12.2022 - 31.12.2023)
 ☒ e-Ärregister - majandusaasta areuannete sisestamine (alates 20.12.2022 - tähtajatu)
 ☐ Statistikaamet - peakasutaja õigused

Märgitud eemaldamisele

×

PRIA: Dokumentide vaataja ⓘ

6.11.2020 - 31.12.2021

[Alusdokument](#)

Märgitud eemaldamisele

×

+ Lisa volitus

Eemalda kõik

Tühista

Kinnita

For each mandate that was selected for deletion - Pääsuke performs this delete request.

2.5.4 Deleting a mandate that has been further sub-delegated

NB! If the mandate to be deleted (by withdrawing it or waiving it) has been further sub-delegated then all the sub-delegated mandates need to be deleted as well inside the same transaction. Pääsuke will not send separate requests to delete these sub-delegated mandates.

The following example illustrates this case.

Let's say Väikefirma OÜ (on 13 December 2023) added a role A to Raamatupidajad OÜ (with the right to further sub-delegate it) and as a result of this a mandate A1 was created.

On 15 December 2023 Raamatupidajad OÜ adds their employee Raili Raamatukoi as a sub-delegate to this mandate A1. As a result, a new mandate A11 is created so that Raili Raamatukoi can now represent Väikefirma OÜ. This mandate A11 must be linked to mandate A1 in the database of the provider of the service. Pääsuke won't store this mandate or the link between A11 and A1 on Pääsuke side.

On 16 December 2023 Raamatupidajad OÜ adds Ülle Pääsuke as a second sub-delegate to this mandate A1. Technically a new mandate A12 is created so that Ülle Pääsuke can now represent Väikefirma OÜ. This mandate A12 must be linked to mandate A1 in the database of the provider of the service.

On 2nd of January Väikefirma OÜ withdraws (võtab tagasi) the mandate A1 that it had given to Raamatupidajad OÜ (or Raamatupidajad OÜ waives (loobub) the mandate) then Pääsuke will send an editMandate (PUT) request for that mandate A1. At the same time, the implementor of the service must also delete (end validity of) mandates A11 and A12, Pääsuke won't send separate delete queries for them. If deleting the mandates A11 or A12 fails on the implementor's side then deleting A1 must fail as well and the error message returned by the implementor will be displayed to the end user.

As the response to the editMandate query the implementor must return the list of mandates that were deleted additionally. The following payload builds on the example described above. The implementator returns the information that mandates A11 and A12 were deleted automatically. ValidityPeriod must reflect the time frame the mandate was actually valid (so 15 December until 2nd of January for Raili and 16 December until 2nd of January for Ülle).

Which (sub-delegated) mandates were deleted additionally

```
{
  "deletedSubDelegatedMandates": [
    {
      "subDelegate": {
        "type": "NATURAL_PERSON",
        "firstName": "Raili",
        "surname": "Raamatukoi",
        "identifier": "EE49302250123"
      },
      "validityPeriod": {
        "from": "2023-12-15",
        "through": "2024-01-02"
      }
    },
    {
      "subDelegate": {
        "type": "NATURAL_PERSON",
        "firstName": "Ülle",
        "surname": "Pääsuke",
        "identifier": "EE48302250123"
      },
      "validityPeriod": {
        "from": "2023-12-16",
        "through": "2024-01-02"
      }
    }
  ]
}
```

2.5.5 Deleting a mandate that has not been further sub-delegated

If the mandate being waived or withdrawn is not further sub-delegated then an empty object must be returned together with HTTP status code 200:

No mandates were deleted additionally (or implementator does not offer adding sub-delegates)

```
{}
```

2.5.6 Digitally signing the request

- if at least one role is being withdrawn or waived (see 2.4.4 for differences) and this role has `RoleDefintion.waivingMustBeSigned=true` or `RoleDefintion.withdrawalMustBeSigned=true` (see chapter 3.3.2 for detailed info) then the user must first digitally sign this request and only after the user has signed then the process will continue and a separate PUT request is sent out for each mandate

2.6 Query addMandate

2.6.1 View to add mandates to a delegate

This is how a new mandate can be added:

Esindaja volitused

×

Esindaja

Sisesta isikukood või registrikood

Esindatav

EMA Aruandja TÜ (EE77543210)

Uue volituse määramine

×

Roll

Maksu- ja Tolliamet: Raamatupidaja pakett ⓘ

Delegeeritavus

☒ Edasivolitamine lubatud

Volituse algusaeg

20.01.2023

Volituse lõppaeg

31.12.2025

☐ Tähtajatu

+ Lisa volitus

Tühista

Kinnita

The user can add several mandates from the same screen. After confirming (by pressing "Kinnita") the flow might continue with the signing step or the adding operations might be performed right away:

- if at least one role to be added has RoleDefinition.addingMustBeSigned (see chapter 3.3.2 for detailed info) then the user must first digitally sign this request and only after the user has signed then the process continues

Pääsuke makes a separate request with each of the mandates to be added:

2.6.2 Path parameters

Query to be sent out by Pääsuke

```
curl -X 'POST' \
  'https://security-server/r1/ee-dev/GOV/70006317/volitused-mock/volitused-estat/v1/representees/EE10391131/delegates/EE38302250123/mandates' \
  -H 'Content-Type: application/json' \
  -d ' <Payload is described in next paragraph> '
```

2.6.3 Payload

Query to be sent out by Pääsuke

```
{
  "representee": {
    Person to be represented
  },
  "delegate": {
    Person getting the representation rights
  },
  "mandate": {
    "role": "GLOBAL1_EMMA:GLOBAL1_EMMA:ACCOUNTANT",
    "canSubDelegate": true,
    "validityPeriod": {
      "from": "2017-07-21",
      "through": "2024-02-21"
    }
  },
  "authorizations": [
    {
      "userIdentifier": "string",
      "hasRoles": "MANAGEMENT_BOARD_MEMBER"
    }
  ],
  "document": {
    "uuid": "5b72e01c-fa7f-479c-b014-cc19efe5b732",
    "singleDelegate": false
  }
}
```

Authorizations are used to show the information about who has confirmed that change and on what grounds (see chapter 3.8).

The presence of "document" indicates that the request was digitally signed (see chapter 3.11).

2.7 Query addSubDelegate

NB! The following is important to only those external parties who plan to offer the functionality of adding sub-delegates.

2.7.1 View where adding a sub-delegate can be started

This flow is started in chapter 2.4 (by pressing the button "Volita edasi") and it can only be started if the "addSubDelegate" link was added.

If the user clicks on that button a view opens up [prefilled](#) with copies of the original mandates:

Esindaja volitused

×

Esindaja

Sisesta isikukood või registrikood

Esindatav

Eesti Volitused MTÜ (EE34567890)

Delegeeritav volitus

×

Roll

Eesti.ee: Töövõimetuslehtede täiendamine ⓘ

Delegeeritavus

Edasivolitamine pole lubatud

Volituse algusaeg

20.01.2023

Volituse lõppaeg

☒ Tähtajatu

Delegeeritav volitus

×

Roll

Eesti.ee: A1 (E101) tõendi ja taotluste toimingud ⓘ

Delegeeritavus

Edasivolitamine pole lubatud

Volituse algusaeg

20.01.2023

Volituse lõppaeg

☒ Tähtajatu

Tühista

Kinnita

The user is allowed to:

- Set the delegate. The delegate can be both a natural person and a legal person.
- Edit the start date as long as it is the same or later than the original start date.
- Edit the end date as long as it is earlier than the original end date. The end date can be set to infinity (tähtajatu) only if the original mandate had the end date set to infinity.
- Delete a role from the list - this means that only a portion of roles get sub-delegated and the role that was deleted from the list will not be sub-delegated.

The user is restricted from:

- Changing the roles
- Adding a new mandate on this screen
- Setting a start date to the past
- Setting a start date to an earlier date than the original start date.
- Setting the mandate to be allowed for sub-delegation (User interface doesn't display this option)
- Setting the end date to be earlier than today.
- Setting the end date to a later value than the original value.

2.7.2 Digitally signing the request

- if at least one role to be sub-delegated has `RoleDefintion.subDelegatingMustBeSigned=true` (see chapter 3.3.2 for detailed info) then the user must first digitally sign this request and only after the user has signed then the process continues

2.7.3 Path parameters

The parameters of the payload are set by the response that loaded the mandates. The parameters are described in paragraph 3.6.3.

2.7.4 Payload

- `subDelegate` is of type `Person` (described in chapter 3.1)
- `validityPeriod` from is only present if it was changed by the user. If it is not present then it must be set to today's date.
- `validityPeriod` through is the last day when the sub-delegated mandate is valid. this cannot exceed the `validityPeriod`->through of the initial mandate.
- `validityPeriod` through without a value (null) means it is valid indefinitely. This is only allowed if the original mandate was valid indefinitely

Query to be sent out by Pääsuke

```
{
  "subDelegate": {
    "type": "NATURAL_PERSON",
    "firstName": "Jüri",
    "surname": "Juurikas",
    "identifier": "EE38302250123"
  },
  "validityPeriod": {
    "from": "2017-07-21",
    "through": "2024-02-21"
  },
  "authorizations": [
    {
      "userIdentifier": "EE39912310123",
      "hasRole": "BR_REPRIGHT:JUHL_SOLEREP"
    }
  ],
  "document": {
    "uuid": "5b72e01c-fa7f-479c-b014-cc19efe5b732",
    "singleDelegate": true
  }
}
```

★ The presence of "document" indicates that the request was digitally signed (see Chapter 3.11). Signing support will be added in Pääsuke version 2.0

2.8 Listing mandates that have been created using adding a sub-delegate (Edasivolitused)

NB! The following is important to only those external parties who plan to offer the functionality of adding sub-delegates.


2.8.1 View "Ettevõtte <ettevõteteni> poolt antud volituste edasivolitused"

In the future, Pääsuke will have a view to see a list of sub-delegators. Let's look at the following screenshot:

Raamatupidajad OÜ (EE23456789)

Ettevõtte esindajad ja volitatud isikud

Ettevõttele antud volitused

 **Väikefirma OÜ (EE12345678)**

Eesti.ee: Töövõimetuslehtede täiendaja ?
Eesti.ee: Töötajate ravikindlustuse andmete vaataja ?

[Volita edasi](#) [Vaata edasivolitusi](#)

The screenshot describes the following situation. Väikefirma OÜ (EE1111111) has given roles "EESTI_EE:TOOVOIMETUSLEHTEDE_TAIENDAJA" and "EESTI_EE:TERVISEKASSA_RAVIKINDLUSTUS" to Raamatupidajad OÜ (EE23456789) with the right to sub-delegate these roles (so Raamatupidajad OÜ can further delegate it to its employees).

Now Raamatupidajad OÜ has sub-delegated (by pressing "Volita edasi") this role to its employee Reijo Raamatukogu

Now a representative of Raamatupidajad OÜ wants to know to whom Raamatupidajad OÜ has sub-delegated these mandates. For that the representative opens tab "Ettevõttele antud volitused" and in the row of "Väikefirma OÜ" he clicks on link "List sub-delegators" (Vaata edasivolitusi).

UI asks the back end to perform the following query to several external parties

CURL query

```
curl -X GET \
  "https://security-server/r1/ee-dev/GOV/70006317/volitused-mock/volitused-estat/v1/representees/EE11111111/delegates/mandates?subDelegatedBy=EE23456789"
```

As mentioned in chapter 2.3.4 it is the same query described in chapter 2.3 with additional query parameter "subDelegatedBy".

The query returns Raamatupidajad OÜ employees Reijo Raamatukogu (EE390121345678) as these mandates were subdelegated by Raamatupidajad OÜ (EE23456789).

If Väikefirma OÜ has given mandates to other parties then they are not returned (since in the query there is "subDelegatedBy" filter parameter in place).

The mandates that are returned have additional property "subDelegator" (that points to the person who created the mandates in the first place by adding a sub-delegate - this is Raamatupidajad OÜ in our case).

Value of "subDelegatorIdentifier" is deprecated but included in the response. If this has a value then the value must equal to the value of subDelegator.identifier.

CURL query

```
{
  "representee": {
    "type": "LEGAL_PERSON",
    "legalName": "Väikefirma OÜ",
    "identifier": "EE12345678"
  },
  "delegate": {
    "type": "NATURAL_PERSON",
    "firstName": "Reijo",
    "lastName": "Raamatukogu",
    "identifier": "EE390121345678"
  },
  "mandates": [
    {
      "role": "EESTI_EE:TOOVOIMETUSLEHTEDE_TAIENDAJA",
      "subDelegable": false,
      "subDelegatorIdentifier": "EE23456789",
      "subDelegator": {
        "type": "LEGAL_PERSON",
        "legalName": "Raamatupidamisfirma OÜ",
        "identifier": "EE23456789"
      },
      "links": {
        "delete": "/EMTA/representees/int-EE12345678/delegates/int-EE390121345678/mandates/int-mandate-123",
      }
    },
    {
      "role": "EESTI_EE:TERVISEKASSA_RAVIKINDLUSTUS",
      "subDelegable": false,
      "subDelegatorIdentifier": "EE23456789",
      "subDelegator": {
        "type": "LEGAL_PERSON",
        "legalName": "Raamatupidamisfirma OÜ",
        "identifier": "EE23456789"
      },
      "links": {
        "delete": "/EMTA/representees/int-EE12345678/delegates/int-EE390121345678/mandates/int-mandate-124",
      }
    }
  ]
}
```

The orange text on the following screenshot indicates how this information will be displayed to the user.

NB! This text is only displayed to the user if subDelegatorIdentifier is set and it is not equal to the person being represented.

Since subDelegatorIdentifier is replaced with subDelegator then the name of the sub-delegator will be made visible as well (this does not yet reflect on the screenshot).

The following example indicates that when the original representee (Väikefirma) is looking at the mandate (that has been created by Raamatupidajad OÜ using adding a sub-delegate) then the orange text is displayed (to avoid confusion for Väikefirma who has created the mandate).

Esindaja volitused

Esindaja

Reijo Raamatukogu
(EE390121345678)

Esindatav

Väikefirma (EE12345678)

Eesti.ee: Töövõimetuslehtede täiendaja ⓘ
EE23456789 lisatud edasivolitus

14.12.2023 - Tähtajatu

Eemalda

Eesti.ee: Puhkusetasu ja keskmise töötasu hüvitamise taotleja ⓘ
EE23456789 lisatud edasivolitus

14.12.2023 - Tähtajatu

Eemalda

+ Lisa volitus

NB! The property mandate.subDlegatorIdentifier should be always added if it is known but Pääsuke might not display it depending of who is viewing the mandate.

2.8.2 View "Isikule <füüsiline või juriidiline isik> antud edasivolitused"

We continue the use-case described in the previous chapter.

Raamatupidajad OÜ has the info that its employee Reijo Raamatukogu has mandates to represent Väikefirma OÜ and these mandates were created by Raamatupidajad OÜ by adding Reijo as a sub-delegate.

Now Raamatupidajad OÜ wants to know where else Reijo Raamatukogu has received access from Raamatupidajad OÜ via sub-delegation.

Raamatupidajad OÜ clicks on his name and system makes a query to find out all the mandates sub-delegated by Raamatupidajad OÜ to Reijo Raamatukogu:

CURL query

```
curl -X GET \
  "https://security-server/r1/ee-dev/GOV/70006317/volitused-mock/volitused-estat/v1/delegates/EE390121345678/delegates/mandates?subDelegatedBy=EE23456789"
```

Systems displays all the mandates that have been delegated to Reio Raamatukogu (personal code EE390121345678) by Raamatupidajad OÜ.

2.9 Support for TÖR (Töötamise Register) use-case

One of the problems Pääsuke wants to resolve is that often times when an employee leaves a company, then nobody will remove the mandates of the person who left. To overcome this issue Pääsuke has taken the following aspect (that has not yet been implemented).

- 11 days after person P has left company C (with EE12345678901) a background process in the employment registry (TÖR) will send a notification to Pääsuke over X-road.
 - This notification is only sent if the person has not been re-registered with company C within this 10-day period

- b. Also there are a lot of other exceptions - if person is still in the management board etc.
- 2. Pääsuke will make queries to detect if the person still has any mandates.
 - a. Pääsuke will make queries "getMandatesByRepresentee" described in chapter 2.3 with the following changes:
 - i. Pääsuke will add one more query parameter ?delegate=EE12345678901 to indicate that Pääsuke is only interested in mandates with this delegate.
 - ii. Pääsuke will not add the header parameter "X-Road-UserId" to this query as this is a query triggered by a background process and not a human being.
- 3. If Pääsuke finds any valid mandates for person P under company C it will send an email to company C informing them about the situation.

3. Types

3.1 Person

Representee or delegate.

Property	Mandatory	Type	Description
type	mandatory	enum(LEGAL_PERSON, NATURAL_PERSON, OTHER, UNKNOWN)	Pääsuke uses this type to display an icon next to the person's name.
firstName	nullable	string	Given names of a natural person. It is not returned together with legalName.
surname	nullable	string	The surname of a natural person. It is not returned together with legalName.
legalName	nullable	string	Legal person name. It is not used together with firstName and surname.
identifier	mandatory	string	See chapter 3.1.1

3.1.1 Person identifier

The maximum length of this property is 256 symbols.

The need for a standard comes from the fact that Pääsuke in its user interface groups together mandates (received from different e-services) of the same person:

 **Jüri Vaarikas** (EE30303039914)

Maksu- ja Tolliamet: **Raamatupidaja pakett** ⓘ
 Statistikaamet (eSTAT): **Andmeesitaja** ⓘ
 Majandusaasta aruannete portaal: **Sisestaja-esitaja** ⓘ
 Asutusteülene: **Raamatupidaja** ⓘ (alates 6.11.2028)

[Muuda](#) [Kopeeri](#) [Eemalda](#)

Each identifier belongs to one of two groups:

1. Two-letter country code (ISO 3166 ALPHA-2) in capital letters followed by person code (see below about person code)
 - a. EE followed by an 8-digit legal entity code from Estonian Business Registry (Äriregister)
 - example: "EE70006317"
 - b. EE followed by an 11-digit national identity number
 - example: "EE60001019906"
 - c. two-letter country code followed by eIDAS identification (1...254 symbols) - this is returned by Tara
 - example: "CZ29d18705-fe88-4b23-9b4c-c073ae12673c"
2. URI - https://en.wikipedia.org/wiki/Uniform_Resource_Identifier . Any valid URI is allowed.
 - a. urn:uuid:{UUID} is recommended if the ID is generated by the party itself. Pääsuke doesn't group such identifiers (for example if Tax and Customs Board and Statistics Estonia return details about a delegate with equal UUID then Pääsuke won't group these records)
 - i. example: "urn:uuid:6e8bc430-9c3a-11d9-9669-0800200c9a66"
 - b. URI for e-mails and phone numbers
 - example: "mailto:John.Doe@example.com" - recommended format for emails (Pääsuke ignores case when grouping)
 - example: "tel:+37251234567" - recommended format for phone numbers
 - c. If two different parties use the same identifiers (for example if Statistics Estonia and Agricultural Registers and Information Board (PRIA) would like to express the same person they would have to agree on common URN)
 - i. urn:{agreed urn value}

3.2 Namespace

Namespace codes are given out by RIA and they cannot contain a slash, colon, semicolon, or space.

3.3 Role and RoleDefinition

3.3.1 Role and role code

This chapter has been moved to a separate document "Introduction to Pääsuke and configuring roles".

3.3.2 RoleDefinition parameters

This chapter has been moved to a separate document "Introduction to Pääsuke and configuring roles".

3.3.3 Representation rights loaded from Business Registry

More information can be found from document "How Pääsuke handles representation rights from the Business Registry (Äriregistrist tulenevad esindusõigused)"

3.4 MandateTriplet

This is called a triplet as it always has 3 components:

Property	Mandatory	Type	
representee	yes	Person (see paragraph 3.1)	The person being represented by the delegate
delegate	yes	Person (see paragraph 3.1)	The person who has the right to represent the representee
mandates	nullable	array Mandate (see paragraph 3.5)	List of mandates that the delegate has for this representee. NB! In order to use reactive processing on both ends the list of mandates in one MandateTriplet is allowed to be up to 100. If one delegate has more than 100 mandates for a representee then additional MandateTriplet(s) must be returned. For example, if the delegate has 121 mandates under a representee then two MandateTriplets would be returned, the first MandateTriplet with 100 mandates and the second MandateTriplet with 21 mandates.

3.5 Mandate

Property	Mandatory	Type	
namespace	yes	namespace code (see paragraph 3.2)	
role	yes	role code (see paragraph 3.3)	
validityPeriod	no	ValidityPeriod (see paragraph 1.7)	
subDelegable	no	boolean	When this mandate was created, was it created with the right to further sub-delegate it (canSubDelegable was true when the mandate was added). When addSubDelegate link is present then subDelegable is always true. On some cases (the user only has viewing rights) it might be that addSubDelegate link is not present but subDelegable is true.
subDelegator	no	Person (see chapter 3.1)	If this mandate was created using sub-delegating then this field points to the (legal or natural) person who had the original mandate. NB! If this mandate was sub-delegated by a legal entity then the field must point to the identifier of the legal entity and it must not point at the natural person (like a board member or that legal entity) who actually carried out the adding of the sub-delegate.
subDelegatorId entifier	no	Person identifier (see chapter 3.1.1)	Same as subDelegator.identifier. This value is deprecated and there is no need to include that in the response.
links	no	MandateLinks (see paragraph 3.6)	links are used to indicate what the user can do with the mandate

3.6 MandateLinks

MandateLinks is a key-value mechanism that allows the provider of the query to indicate what actions can be done with the mandate in the Pääsuke UI.

The list of properties is fixed but new keys might be added over time.

The value of each property has to follow a pre-defined format, but the format lets the provider of the query use identifiers inside the value

All the keys of this type are nullable so if some action is not supported by the mandate then the corresponding value of the key is null (or not included at all in the response).

Property name	Format of the value NB! Everything that is not surrounded by curly brackets is fixed.
delete	/representees/{representeeld}/delegates/{delegateld}/mandates/{mandateld}
addSubDelegate	/representees/{representeeld}/delegates/{delegateld}/mandates/{mandateld}/subdelegates
origin	Reserved for future. Pääsuke would open a new window/tab for the USER with this URL to view the mandate at its origin.

3.6.1 delete

If this property is present with a non-null value it indicates that the mandate can be deleted using Pääsuke (this means withdrawing a mandate or waiving a mandate).

If the property is missing or null then Pääsuke forbids the user from removing this mandate.

If the user confirms removing this mandate from Pääsuke then Pääsuke sends out the deleteMandate (see paragraph 2.6) query using parameters parsed from the value.

The value of the "{ns}" has to match the namespace of the role of the mandate.

Fragment of example output

```
"links": {
  "delete": "/v1/representees/1234/delegates/5678/mandates/901234"
}
```

When the user decides to delete the mandate from Pääsuke then Pääsuke sends out the following query to the same party that returned the response.

So if the mandate to be deleted was served to Pääsuke by "ee-dev/GOV/70006317/volitused-mock/volitused-empta" then Pääsuke sends out the following query:

Query to be sent out by Pääsuke

```
curl -X 'DELETE' 'https://security-server/r1/ee-dev/GOV/70006317/volitused-mock/volitused-empta/representees/1234/delegates/5678/mandates/901234'
```

3.6.2 addSubDelegate

If this property is present with a non-null value it indicates that the mandate can be further sub-delegated.

If the property is missing or null then Pääsuke forbids the user from sub-delegating this mandate.

If the role definition metadata states that the role cannot be sub-delegated then Pääsuke forbids the user from sub-delegating this mandate even if this property is present in the output.

If the user sub-delegates this mandate in Pääsuke then Pääsuke sends out the addMandateSubDelegate (see paragraph 2.7) query using parameters parsed from the value.

The value of the "{ns}" has to match the namespace of the role of the mandate.

Fragment of example output

```
"links": {
  "addSubDelegate": "/representees/R987/delegates/D654/mandates/M321/subdelegates"
}
```

When the user adds a sub-delegate then

So if the mandate to be deleted was served to Pääsuke by "ee-dev/GOV/70006317/volitused-mock/volitused-ema" then Pääsuke sends out the following query:

Query to be sent out by Pääsuke

```
curl -X 'POST' \
  'https://security-server/r1/ee-dev/GOV/70006317/volitused-mock/volitused-ema/GLOBAL1_EMTA/representees/EE10391131/delegates/EE38302250123/mandates/M321/subdelegates' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' payload of the message is described in paragraph 2.7
```

3.6.3 update

This is reserved for the future to update the record's validity period and the record's boolean flag "can add sub-delegate".

3.6.4 origin

URL to self-service where the mandate information can be displayed to the user.

This is reserved for the future.

3.7 Translation

Pääsuke runs within eesti.ee portal that is offered to end users in Estonian, English, and Russian.

Returning translations in Estonian is mandatory.

If the English or Russian translation is missing then for that part the user interface of Pääsuke uses the Estonian translation instead.

Property name	Mandatory	Type	Description
et	yes	string	Translation in Estonian
en	no	string	Translation in English
ru	no	string	Translation in Russian

3.8 Authorization

This list is added to some of the payloads to reflect the information on why the person doing a modification was allowed by Pääsuke to perform the action.

There are several options

3.8.1 Person has a role that allows creating, sub-delegating, and removing mandates

```
"authorizations": [
  {
    "userIdentifier": "EE49028099999",
    "hasRole": "STAT:Peakasutaja"
  }
]
```

3.8.2 Person is on the board and has the right to represent the legal entity alone

```
"authorizations": [
  {
    "userIdentifier": "EE49028099999",
    "hasRole": "BR_REPRIGHT:JUHL_SOLEREP"
  }
]
```


3.8.3 Several people who are on the board and they have partial rights (ühisesindusõigus) but together they can represent the legal entity

```
"authorizations": [
  {
    "userIdentifier": "EE49028099999",
    "hasRole": "BR_REPRIGHT:JUHL_GROUPREP"
  },
  {
    "userIdentifier": "EE39211110000",
    "hasRole": "BR_REPRIGHT:JUHL_GROUPREP"
  }
]
```

This use case (ühisesindusega juhatuse liikmed saavad Pääsuke abil rolle peale panna) is not yet supported but the API is already designed to be able to support it in the future

3.9 ValidityPeriod

Property	Mandatory	Type	Description
from	nullable	date	The first day (inclusive). Can be both in the past and in the future.
through	nullable	date	The last day (inclusive). If the value is missing (or null - sending nulls is discouraged) it means the end date is not specified (infinity). Normally this date can never be in the past (as Pääsuke only returns mandates that are currently valid or become valid in the future).

3.10 Problem

<https://www.rfc-editor.org/rfc/rfc7807>

<https://blog.axway.com/learning-center/apis/api-design/introduction-to-rfc-7807>

Property	Mandatory	Type	Description
type	no		An absolute URI that identifies the problem type
href	no		An absolute URI that, when dereferenced, provides human-readable documentation for the problem type (e.g. using HTML).
title	yes		A short summary of the problem type. Written in English and readable for engineers (usually not suited for non-technical stakeholders and not localized). Example: Service Unavailable
status	no		This reflects the HTTP status code and is a convenient way to make problem details self-contained. That way they can be interpreted outside of the context of the HTTP interaction in which they were provided
translation	no	Translation (chapter see 3.4)	A human-readable description of the problem <i>instance</i> , explaining why the problem occurred in this specific case. This value could and often will be displayed to the user.
ticket	no		ticket number
<future attributes>	no		adding other attributes is allowed

Returns list, but the system only reads the first element.

3.11 Document

★ - This section is present if the persons listed in the authorizations block added their digital signature. Signing support will be added in Pääsuke version 2.0

Property	Mandatory	Type	Description
uuid	yes	string	Document UUID. This identifier can be used to download the signed container from Pääsuke over X-road.

singleDelegate	yes	boolean	This is set to true in case the signed document contains information about a single delegate and it is safe to reveal the document to the delegate. False means that the signed container contains information about multiple delegates and it is only safe to reveal it to the representee and not to any of the delegates.
----------------	-----	---------	--

4. Details about roles and namespaces

4.1 Namespaces with parent-child relation

RIA defines global roles (in Estonian *katusrollid*) that can be supported by multiple institutions.

The idea is to provide all needed permissions in different institutions for a job as one single access role.

Let's look at this using an example. RIA has created a parent namespace "GLOBAL1" with the following configuration

Definition of a namespace		
namespace	GLOBAL1	
type	PARENT	
parent namespace	<null>	
title in Estonian	Asutusteülene	NB! This is a draft name for the role
title in English	Across institutions	NB! This is a draft name for the role
owner	GOV/70006317	This refers to Riigi Infosüsteemi Amet. The owner refers to the party who declares roles in this namespace.

This namespace contains the following roles:

Definition of roles			
namespace	GLOBAL1	GLOBAL1	GLOBAL1
role code	GLOBAL1:Accountant	GLOBAL1:Human_Resources_Specialist	GLOBAL1:Data_viewer
title in Estonian	Raamatupidaja	Personalitöötaja	Andmete vaataja

However, it is not possible for anyone to directly grant roles to some delegate from such a parent namespace (as any role of type parent is an interface that declares some properties for any child roles).

Any institution can decide to support all or only a selection of these roles. Let's say the Estonian Tax and Customs Board (EMTA) has added support for all 3 roles.

For this Pääsuke has defined a separate namespace GLOBAL1_EMFTA for EMTA:

Definition of a namespace	
namespace	GLOBAL1_EMFTA
type	CHILD
parent namespace	GLOBAL1
owner	GOV/70000349

In this namespace there are definitions of the same 3 roles with the same names for the roles (descriptions are different):

Definition of roles			
namespace	GLOBAL1_EMFTA	GLOBAL1_EMFTA	GLOBAL1_EMFTA
role (allowed values are fixed by roles in the parent namespace)	GLOBAL1_EMFTA:Accountant	GLOBAL1_EMFTA:Human_Resources_Specialist	GLOBAL1_EMFTA:Data_viewer
title (title must be set but it is ignored by Pääsuke)	Raamatupidaja	Personalitöötaja	Andmete vaataja
description in Estonian	samad õigused, mis raamatupidaja pakettis	Töötajate registri kasutamise õigused	Andmete vaataja õigused

If there is some other institution (let's say Statistics Estonia (from now on STAT) that only wants to add support for the role "Accountant" then a separate child namespace will be created for that (that points to GLOBAL1 parent namespace) and STAT only declares the Accountant role in that namespace:

Definition of a namespace	
namespace	GLOBAL1_STAT
type	CHILD
parent namespace	GLOBAL1
owner	GOV/70000332

Definition of roles	
namespace	GLOBAL1_STAT
role	GLOBAL1_STAT:Accountant
title (title values of global child roles must be set but are ignored by Pääsuke)	Raamatupidaja
description in Estonian	Peakasutaja õigused

The following screenshot demonstrates how the Accountant role will be visible to the end user who starts to assign the Accountant role to some employee:

The screenshot assumes there are 4 agencies that support that role:

Esindaja volitused

Esindaja

Ülle Pääsuke (46414160202)

Esindatav

Väikefirma OÜ (EE12345678)

Uue volituse määramine

Roll

Asutusteülene: Raamatupidaja

Volituse algusaeg

Niipea kui võimalik

Volituse lõppaeg

☒ Tähtajatu

Rolli kehtivus asutustes:

☒ Maksu- ja Tolliamet - samad õigused, mis raamatupidaja pakettis
☒ Eesti.ee - töövoimetustele kinnitamine
☒ e-Ariregister - majandusaasta aruannete sisestamine
☒ Statistikaamet - peakasutaja õigused

+ Lisa volitus

Tühista

Kinnita

The person who is assigning the role can untick some of the institutions. Eventually, when the person clicks confirm (Kinnita) Pääsuke performs one POST request to each institution over the x-road to add the role for each institution. So if the person didn't untick anything then Pääsuke will make 4 separate post requests to add 4 separate roles.

4.2 Special parent-child namespaces for Admin Portal access rights

NB! This chapter is relevant for parties who keep mandates in Pääsuke.

Let's say some government agency (Agency-X) keeps all the mandates in Pääsuke and queries them from there.

Now some person who is a management board member of some company (Company-C) sends a digitally signed request and asks for a mandate to be added for the employee (Employee-E) of his company.

The administrative user of that agency (after verifying that the management board member has the right to represent that company) needs an administrative user interface to enter this mandate into Pääsuke.

For this Pääsuke is going to offer X-road services that the agency can use using the MISP2-based Admin Portal of that agency.

There is a need for a mechanism for that agency to declare which of its roles can be assigned or deleted via Admin Portal.

For this RIA has created a parent namespace namespace "ADMIN_PORTAL" and added one role "ADMIN_USER" into it.

The agency (Agency-X) that wants its administrative person to add the mandates from Admin Portal portal needs to declare a child namespace:

ADMIN_AGENCYX and adds a role into it (ADMIN_AGENCYX:ADMIN_USER).

And then the agency is free to modify the addableBy and withdrawableBy properties (described in the chapter 3.3.1) of that role:

For example, the following declares a role that only Admin Portal users can assign or delete:

addableBy: ["ADMIN_AGENCYX:ADMIN_USER"]

withdrawableBy: ["ADMIN_AGENCYX:ADMIN_USER"]

4.3 Dedicated namespaces

These namespaces are used in Authorization (see Chapter 3.8) and also in RoleDefinition as values in addableBy, withdrawableBy, waivableBy, subDelegableBy.

4.3.1. Dedicated namespace for representation rights loaded from Business Registry (Äriregistrist)

BR_REPRIGHT is a namespace in Pääsuke that indicates that the representation rights are loaded by Pääsuke from Business Registry.

4.3.2. Dedicated namespace for mandates managers

PAASUKE is a namespace in Pääsuke for creating mandate manager roles.

4.3.3. Dedicated namespace for mandates managers

PAASUKE_ADMIN is a namespace in Pääsuke for creating superusers.

4.3.4. Dedicated namespace for representation rights between natural persons

NAT_REPRIGHT is a namespace in Pääsuke that indicates that the natural person has a right to represent (esindaja teovõime ei ole piiratud) another natural person (that could be oneself).