

Version: 1.7.1
Date: 2020-05-12



evry.com

[Classification]

Documentation

EPH REST API Vedlegg AutopassApi

Revision History

Revision no.	Revision date	Changes	Revised by
0.9	15.2.19	Initial version, distributed and confirmed by AutoPASS Architecture Forum	EVERY
1.0	27.05.19	Implementation of Security tokens to be aligned with security standard in AutoPASS	
1.1	26.06.19	Some clarification of responses	Jo Inge Stubbe
1.2	06.08.19	Removed field for "Pricing correction" Added field for ImageTransactionID	Jo Inge Stubbe
1.3	27.08.19	24 hours rule for claim status 4, reason code 10	Jo Inge Stubbe
1.4	05.11.19	Added field for ExitStationActorID and PaymentDate	Dan Isak Kveta
1.5	26.11.2019	Revision for PaymentDate and general cleanup	Jonathan Murray
1.6	26.11.2019	Changed EventDate to PaymentDate to add clarity	Jonathan Murray
1.7	02.04.2020	Corrected and updated outdated information.	Robin Berg Pettersen
1.7.1	12.05.2020	Minor correction on TransitResults -> TransitResult, updates on the TransitResult format, and other minor corrections related to the transits/ endpoint.	Robin Berg Pettersen

Contents

1	EPH integration	4
1.1	Background	4
1.2	Security	5
1.3	REST API - Token	6
1.4	REST API - Transits	7
1.5	Transits output field descriptions	8
1.6	REST API - TransitResponses	11
1.7	TransitResponses POST field descriptions	13
1.7.1	ClaimStatus codes	13
1.7.2	Reason codes	13
1.7.3	Error handling	14

1 EPH integration

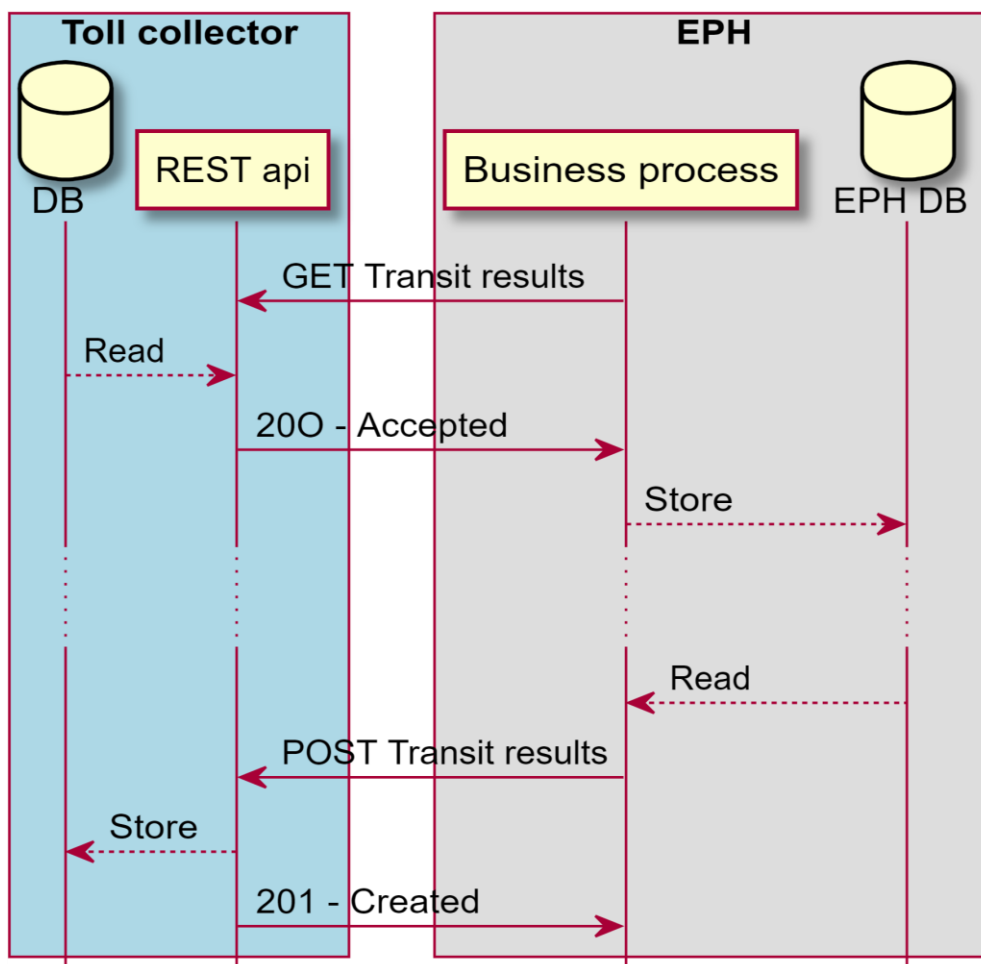
1.1 Background

This document describes the integration between Norwegian toll collectors and the EPH.

A simple REST API (**transits**) hosted by the TC will provide transit data to the EPH. The EPH operator will post its response back to a similar REST API (**transitreponse**).

When EPH later receives partial or full payment for a transit, this payment along with the original transit Id must be posted back to the Toll collector using the /transitresponses API. The amount of payment received (including VAT) must be included in the JSON object, to set the amount which TC will then show on the invoice to EPH.

The EPH will be responsible for pulling all transit data, and to respond to each transit with a transit response. If the EPH is polling for transit data that it has already responded to, that response will follow the transit data.



1.2 Security

Consumers of the API request a token from the endpoint '/token', with user data (username, password) to authenticate and authorize the API client. For this endpoint a parameter grant_type with the value of "password" is also required.

The username that is used in the request will be checked against the customer register in the toll collector's system to ensure that the user request/response is connected to a valid EPH collection company.

On successful request the response (HTTP 200) is a token with a default time to live of two minutes. The token must be added as a request header 'Authorization' Bearer <token>. If not authenticated, the response is Access Denied (HTTP 401)

Use of https guarantees integrity of the transmitted data, so there will be no need for any manual measures to enforce integrity.

1.3 REST API - Token

Url:	/token/
Method:	POST
Header:	
Content-Type	application/x-www-form-urlencoded
Success response:	200 - Accepted
Error response:	401 – Access denied
	403 - Forbidden
	500 - Internal Server Error

Example:

POSThttps://www.yourautopassapiendpoint.no:4443/AutopassApi/tokenSend

Form3AuthQueryHeader1Docs

usernametheuser

passwordpassword123

grant_typepassword

New nameNew value

200 OK31 ms357 B2 Minutes Ago

PreviewHeader8CookieTimeline

1 {

2 "access_token": "vUI6nyZbpL52fotA0c9hJ8aYchulDKA91RTcjzT-DuTqt33WwjN4MmfTtnCaEgaDVe2vDEi6Hf1FtghJ6-kzadxbUmMq-j0Rd9Mwf7sdTvS3JUNEHC-0e6MKf731hMbFlA6Djosdqa_xtwL0qfjV17-6EiMUmxRQgIGWdU-hPkz7KWtILhrAhEE6nkXkelW5TZ8r8RoEnJ0473WYALV21491dkagQVfitLuHd0dbanahvteMTQobK5n-bw4mKneCAFp6wIqIackJ0sMPi19I_Q0WtnVcCd9shWj50Zko",

3 "token_type": "bearer",

4 "expires_in": 119

5 }

Name	Description
access_token	The generated token key that must be passed to all calls done to the API. When it expires another token must be generated. The expiry time is configured by the API-configuration and has a default lifespan of two minutes.

Page 6/14

Sensitivity: Internal

token_type	What kind of token used. This API uses Bearer Token.
expires_in	How long the token will live in seconds.

1.4 REST API - Transits

api/{tcActorId}/transits/?from={from including}&to={to less than}

Url: Example:
https://www.yourautopassapiendpoint.no:4443/AutopassApi/api/999999/transits/?from=2018-08-09T00:00:00&to=2020-03-20T10:12:13

Method:
GET

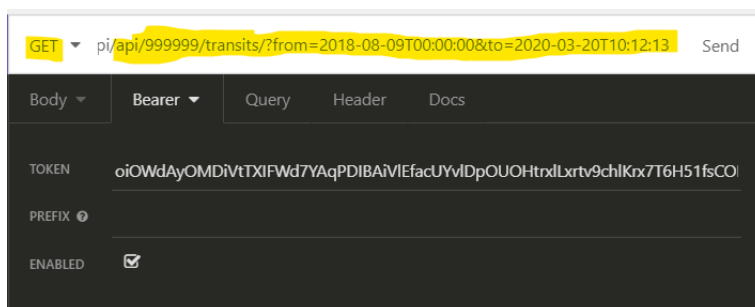
Header:
Authorization: Bearer Token

Params:
"from" mandatory (date time) format: yyyy-MM-ddTHH:mm:ss
"to" mandatory (date time) format: yyyy-MM-ddTHH:mm:ss

Success response:
200 – OK
204 – No Content

Error response:
400 - Bad Request
401 – Access denied (e.g. bearer token is not specified or out of date)
403 – Forbidden (valid login but account does not have the required access to use this request)
500 - Internal Server Error

Example for request:



Body:

```
{
  "Transits": [
    {
      "Id": "26093266-000f-4c35-a0dd-4582e21b7db4",
      "TypeOfTransit": "Cx",
      "TimeOfTheExitTransit": "2018-12-07T15:48:22",
      "ExitStationStationCode": 1234,
      "ExitStationActorID": 4321,
      "LaneIdentification": "A001",
      "TariffClassification1": "B1",
      "VehicleClass": 0,
      "FeeVATExcluded": 20.0,
      "FeeVATIncluded": 20.0,
      "LicensePlateNumberDetect": "ABC123",
      "NationLicensePlateDetect": "SE",
      "TCTransactionIdent": "1234567890123456",
      "ImageTransactionID" : "100007001012018102517592300000020",
      "TransitResult": [
        {
          "ClaimStatus": 2,
          "ReasonCode": 0,
          "PaymentDate": "2019-01-02T00:00:00",
          "Amount": 15.0,
          "InvoiceNo": 5324323
        },
        {
          "ClaimStatus": 2,
          "ReasonCode": 0,
          "PaymentDate": "2019-02-03T00:00:00",
          "Amount": 5.0,
          "InvoiceNo": 0
        }
      ]
    }
  ]
}
```

1.5 Transits output field descriptions

Field	Description
Id	Unique identifier for the transit, set by the Toll Collector

Field	Description
TypeOfTransit	Indicates different characteristics of the transit: Cx/Dx/Ex = Debit transactions and Rx/Sx/Tx = Credit transactions, Cx/Rx primary transactions, Dx/Sx used if transaction has amount with divided VAT or VAT is calculated in one line, Ex/Tx used for aggregated transactions. TypeOfTransactions that is usually used is C4, D4 for passages
ExitStationStationCode	Station code for the toll station the transit occurred at
ExitStationActorID	Unique ID for each toll charger.
LaneIdentification	Lane number at the specified toll station
TariffClassification	TC specific vehicle class applied for tariff calculation
VehicleClass	As retrieved from the OBE during transaction; filled with zeroes in case of measured classification (Class from OBE)
FeeVATExcluded	Fee associated to the transaction (in the smallest unit, e.g. euro cents if euro is the currency)
FeeVATIncluded	Fee associated to the transaction (in the smallest unit, e.g. euro cents if euro is the currency) In case of aggregated VAT calculation the C tx shall be filled with 0
LicensePlateNumberDetect	The detected license plate (may have been detected automatically or manually)
NationLicensePlateDetect	The detected nationality of the license plate
TCTransactionIdent	Identifier for the transaction lines which shall be seen as one transaction. E.g. When Cx/Dx are part of the same transaction they have the same number. When discount or corrected price result in several lines they must be identified with the same number.
ImageTransactionID	Unique ID for a passage's related images. ImageTransactionID is build based on relevant information for the passage.
TransitResult	List of TransitResult JSON objects with the following fields:
ClaimStatus	EPH claim status codes. See table "Claim status codes"
ReasonCode	EPH claim status reason. A code accompanying the claim status. See table "Reason codes"
PaymentDate	EPH's bank date (if claimStatus 1 or 2). The format of the date is JSON datetime (yyyy-MM-ddTHH:mm:ss). If the date is blank, the value will be "1900-01-01T00:00:00".
Amount	Amount

Field	Description
InvoiceNo	The invoice number for the payment. If the payment has yet to be invoiced, this field will show the value 0.

1.6 REST API - TransitResponses

	api/{tcActorId}/transitresponses
Url:	Example: https://www.yourautopassapiendpoint.no:4443/AutopassApi/api/999999/transitresponses
Method:	POST
Header:	Authorization: Bearer Token
Params:	None
Success response:	201 - Created
Error response:	400 - Bad Request 401 – Access denied (e.g. web token is not specified or out of date) 403 – Forbidden (valid login but account does not have the required access to use this request) 500 - Internal Server Error

Example POST Body:

```
{
  "TransitResults": [
    {
      "Id": "ad33f3a8-8209-4077-8c31-b5400d0329a8",
      "ClaimStatus": 2,
      "ReasonCode": 0,
      "PaymentDate": "2019-01-02T00:00:00",
      "PaidAmountVATIncluded ": 15.0
    },
    {
      "Id": "d1d2abe6-bbea-4655-8160-e130c8ec6a75",
      "ClaimStatus": 2,
      "ReasonCode": 0,
      "PaymentDate": "2019-02-03T00:00:00",
      "PaidAmountVATIncluded ": 5.0
    },
    {
      "Id": "14a3d336-1183-46c9-9cfc-189f09c158d2",
      "ClaimStatus": 3,
      "ReasonCode": 01,
      "PaymentDate": "1900-01-01T00:00:00",

```

```
    "PaidAmountVATIncluded ": 0.0  
  }  
]  
}
```

1.7 TransitResponses POST field descriptions

Field	Description
TransitResults	List of TransitResult JSON objects with the following fields:
Id	GUID for EPH Transit ID. This is the key GUID which identifies the transit and is set by the Toll collector. This value must be entered to allow the application to identify the transit. Mandatory.
ClaimStatus	EPH claim status codes. See table "Claim status codes". Mandatory.
ReasonCode	EPH claim status reason. A code accompanying the claim status. See table "Reason codes". Mandatory.
PaymentDate	EPH's bank date (if claimStatus 1 or 2). The format of the date is JSON datetime (yyyy-MM-ddTHH:mm:ss). If there is no payment date the value "1900-01-01T00:00:00" must be used. Mandatory.
PaidAmountVATIncluded	EPH's bank date (if claimStatus 1 or 2). The format of the date is JSON datetime (yyyy-MM-ddTHH:mm:ss). If the date is blank, the value will be "1900-01-01T00:00:00". Mandatory.

1.7.1 ClaimStatus codes

ClaimStatus	Description
1	Full payment
2	Partial payment
3	No payment
4	Request for correction

1.7.2 Reason codes

ClaimStatus	ReasonCode	Description
3	01	Unable to get vehicle owner identification
3	02	Unable to collect payment from vehicle owner
4	04	Registration number is incorrect
4	05	Country code for registration number is incorrect

4	10	Vehicle information is incorrect. Shall not be called before 24 hours after new HGV file is updated
---	----	---

1.7.3 Error handling

If the request from the web client is invalid, the API will return an http error (for example 400 – Bad request) and a JSON object with details of the error. The error handling mechanism outlined by the Internet Engineering Task Force here <https://tools.ietf.org/html/rfc7807> is implemented in this solution.

If the accept header of a request contains “application/problem+json”, any error returned will declare that as its content type. If not, errors will be returned with normal JSON as it’s declared content type, though the contents will be the same regardless. The following is an example of an error response body:

```
{
  "Type": "relativeurl/errors/error-type/",
  "Title": "Error title",
  "Status": 400,
  "Details": "Longer description of the error."
}
```

1.7.3.1 Possible error situations

The following errors can occur as a result of illegal parameter combinations in a request:

- Http Status Code 400:
 - Illegal or malformed date-values. E.g. future dates, or dates other than the default for Claim Statuses 3 and 4
 - Undefined Claim Statuses
 - Malformed body in a POST request
- Http Status Code 500:
 - The server failed to process part of the request
 - The server crashed unexpectedly
 - The system failed to properly retrieve account information for a valid login id
- Http Status Code 401:
 - Invalid login credentials
 - Invalid or expired bearer token
- Http Status Code 403:
 - The login id was valid, but the account does not have access to any parts of the system due to missing account-configuration
 - The login was valid, but system access was revoked after login