**Styrelsen for dataforsyning og effektivisering – Ledningsejerregistret**

**Postman Guide**

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

|  |  |
| --- | --- |
| **Version:** | 1.0 |
| **Status:** | Final |
| **Approver:** | Niels Nepper |
| **Author:** | Jonathan Nygaard Magnussen |



**Document history**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | Date | Author | Status | Comments |
| 1.0 | 30-03-2020 | Jonathan Nygaard Magnussen | Draft |  |
| 1.0 | 08-04-2020 | Rasmus Ringsborg Howalt Owe | Final | Review. |

**References**

|  |  |  |  |
| --- | --- | --- | --- |
| Reference | Title | Author | Version |
| **[C0200 – Webservice guide]** | [**C0200 - Guide to Web Services**](https://goto.netcompany.com/cases/GTE490/SDFELER/Deliverables/C0200%20-%20Guide%20to%20Web%20Services.docx) | **Netcompany** | **Seneste** |
| **[C0200 – Udfyldelse af GML]** | [**C0200 - Vejledning til udfyldelse af GML for udveksling af ledningsoplysninger**](https://goto.netcompany.com/cases/GTE490/SDFELER/Deliverables/C0200%20-%20Vejledning%20til%20udfyldelse%20af%20GML%20for%20udveksling%20af%20ledningsoplysninger.docx) | **COWI** | **Seneste** |

Table of contents

[1 Introduction 3](#_Toc37758060)

[2 Postman 3](#_Toc37758061)

[2.1 Installing Postman 3](#_Toc37758062)

[2.2 Collection and environments 3](#_Toc37758063)

[2.3 Postman certificates 6](#_Toc37758064)

[2.4 Using Postman 9](#_Toc37758065)

[3 Example 10](#_Toc37758066)

[3.1 Using Hent Anmodninger (integration 10) 11](#_Toc37758067)

[3.2 Using Kvitter for Anmodning (integration 11) 11](#_Toc37758068)

[3.3 Using Send Graveforespørgselssvar (integration 13) 12](#_Toc37758069)

# Introduction

This document aims to describe how the new LER 2.0 web API can be tested using Postman. Therefore, the document is intended for people with technical knowledge that are to implement an integration to LER 2.0.

The Examples in this document are intended to be used as a demonstration and can contain account-specific or time-dependent properties. In consequence, any displayed values are purely meant for displaying how and where to enter values, and the user should provide their own valid data.

# Postman

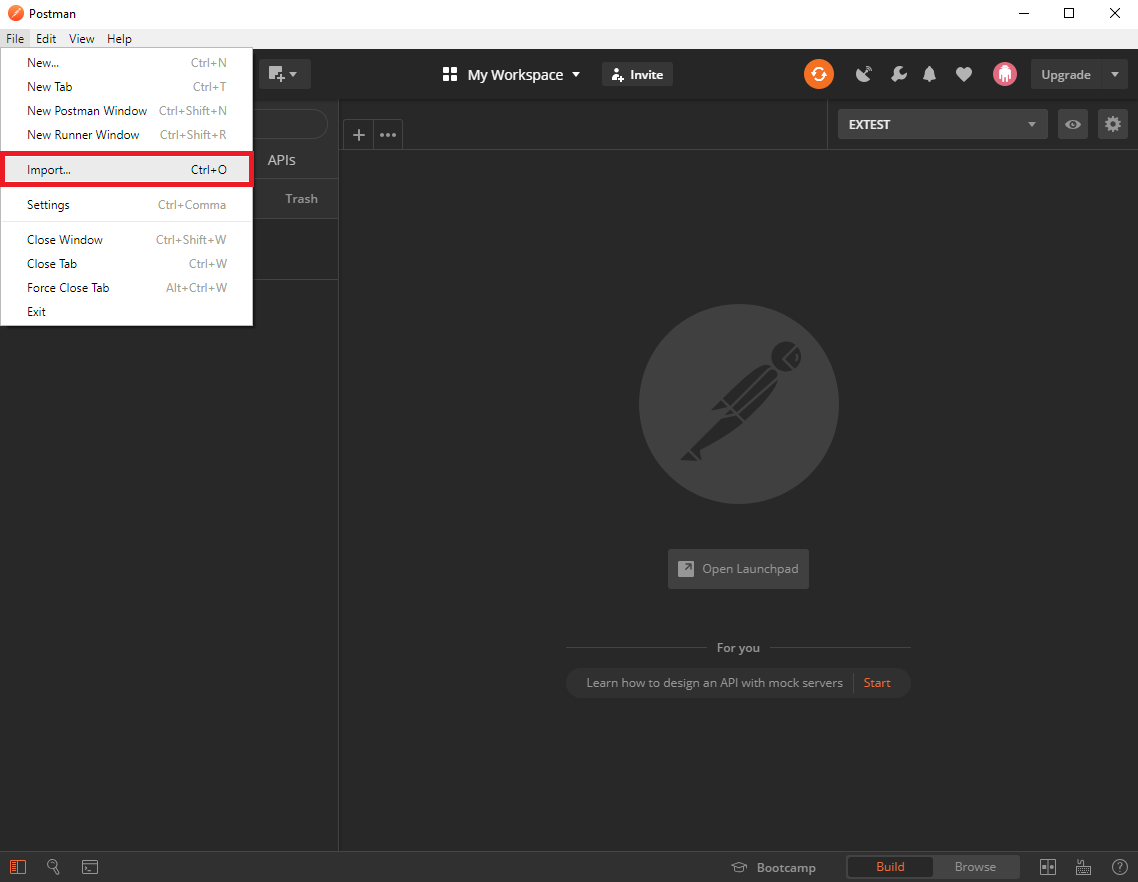
## Installing Postman

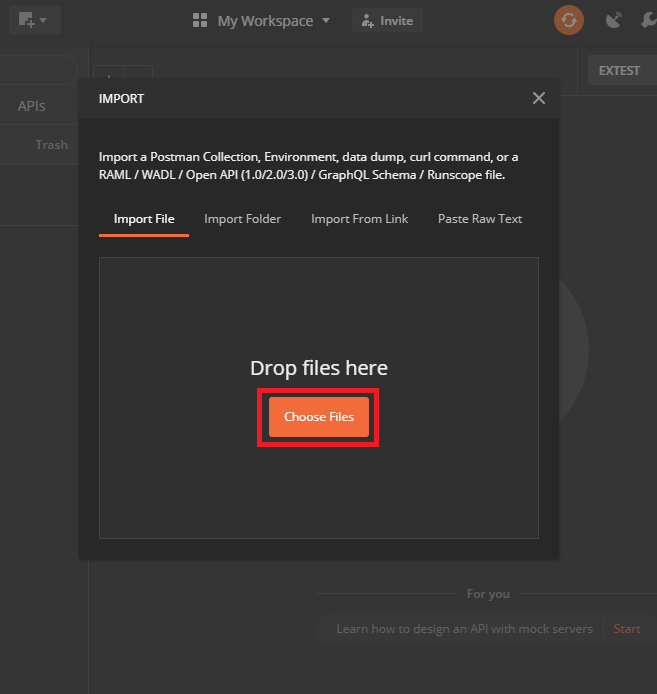
This guide uses Postman version 7.20.1. Postman can be downloaded from <https://www.postman.com/downloads/>, and installed by using the default options.

## Collection and environments

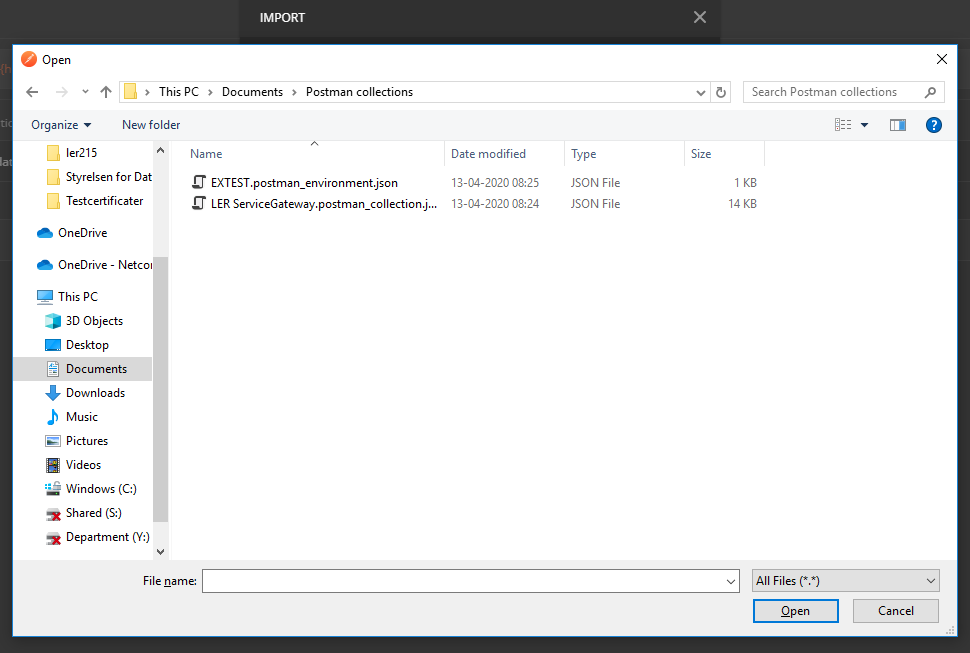
The next step after Postman has finished installing, is to set up Postman with the correct collection of calls and environments. This can be found in the accompanying folder.

The following step-by-step guide shows how to import a collection to Postman:

1. Open Postman.
2. Select “File” -> “Import…”.
3. In the Import popup, select “Choose Files”, or drag the files from the folder mentioned in step 4 directly onto the popup.



1. Navigate to the destination of the collection folder.
2. Select all the files.



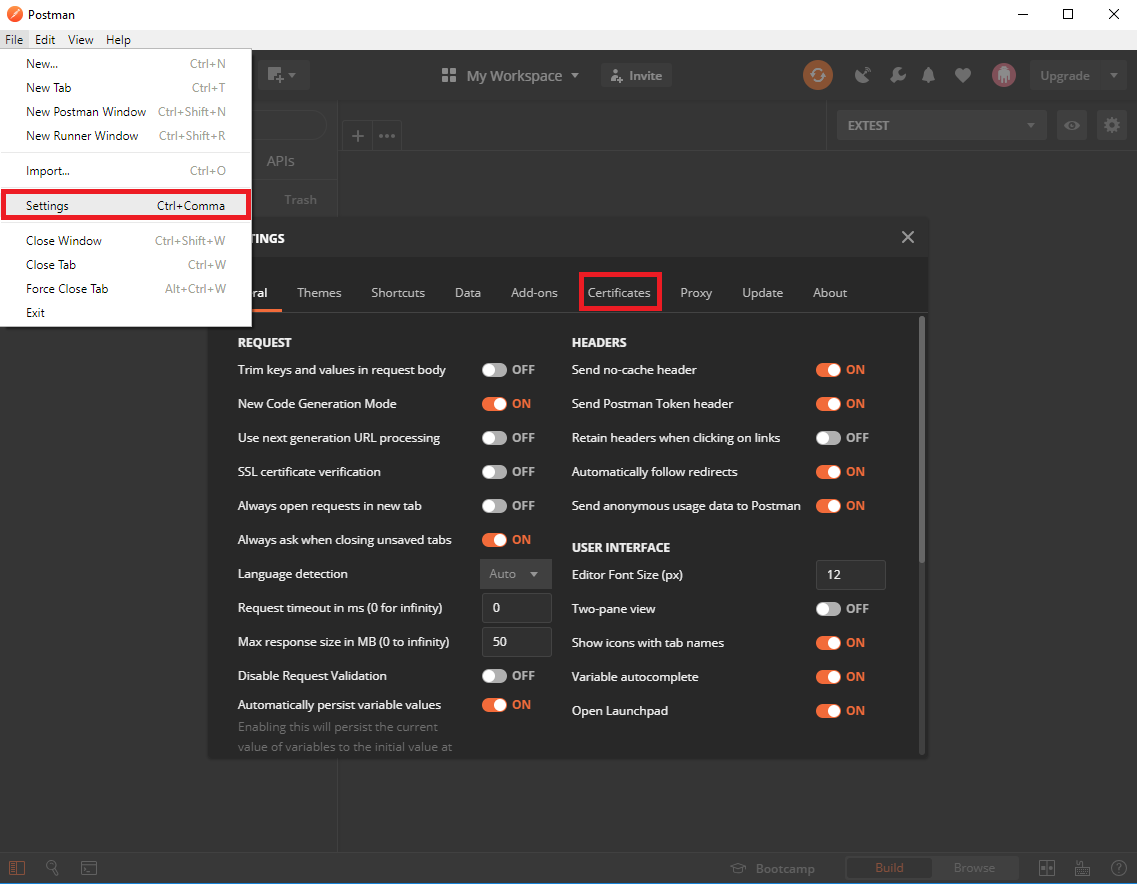
1. Press “Open” to finish importing the collection and environments.

## Postman certificates

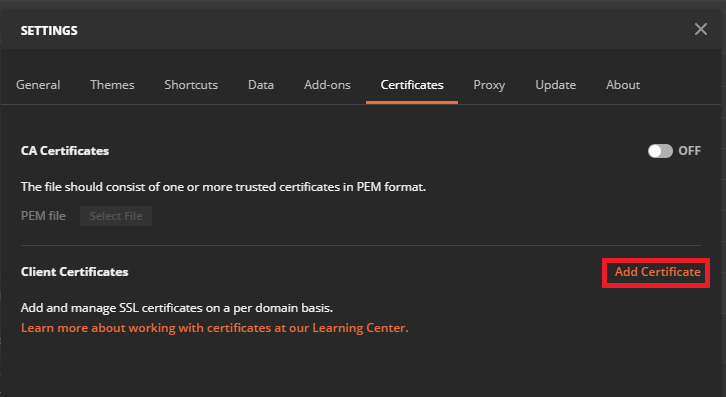
Before postman can be used to call LER’s REST services, an accepted FOCES certificate must be defined for each of the hosts. Certificates are note provided, and should be obtained by the user.

The following step-by-step guide explains how to add certificates in Postman:

1. Go to “File” -> “Settings”, and navigate to “Certificates”.



1. Within “Certificates” is a list of added certificates and their related hosts. Click “Add Certificate” to add a new certificate to the list.



1. Fill out the form as shown below, using host values from Figure 1 below. Click “Add” to add the certificate
   1. Insert host: services-extest.ler.dk.
   2. Insert port: 443.
   3. Provide valid certificate PFX file.
   4. Enter password for the chosen PFX file.

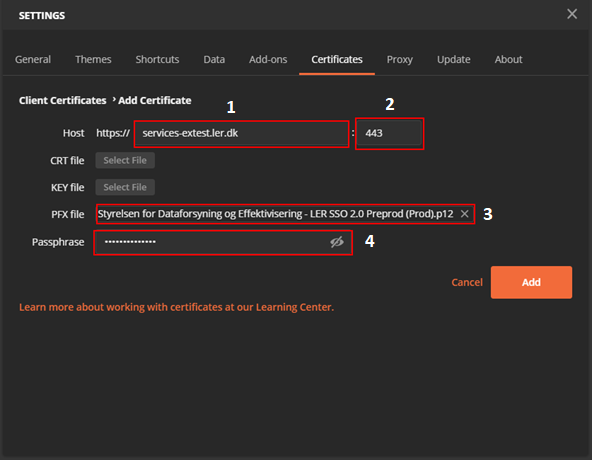


Figure 1 Adding new certificates to Postman.

## Using Postman

Once Postman has been set up, it should now be possible to call LER’s 2.0 web API.

Figure 2 shows the list of possible calls to LER, for a more in-depth description of each of these calls see [C0200 – Webservice guide].

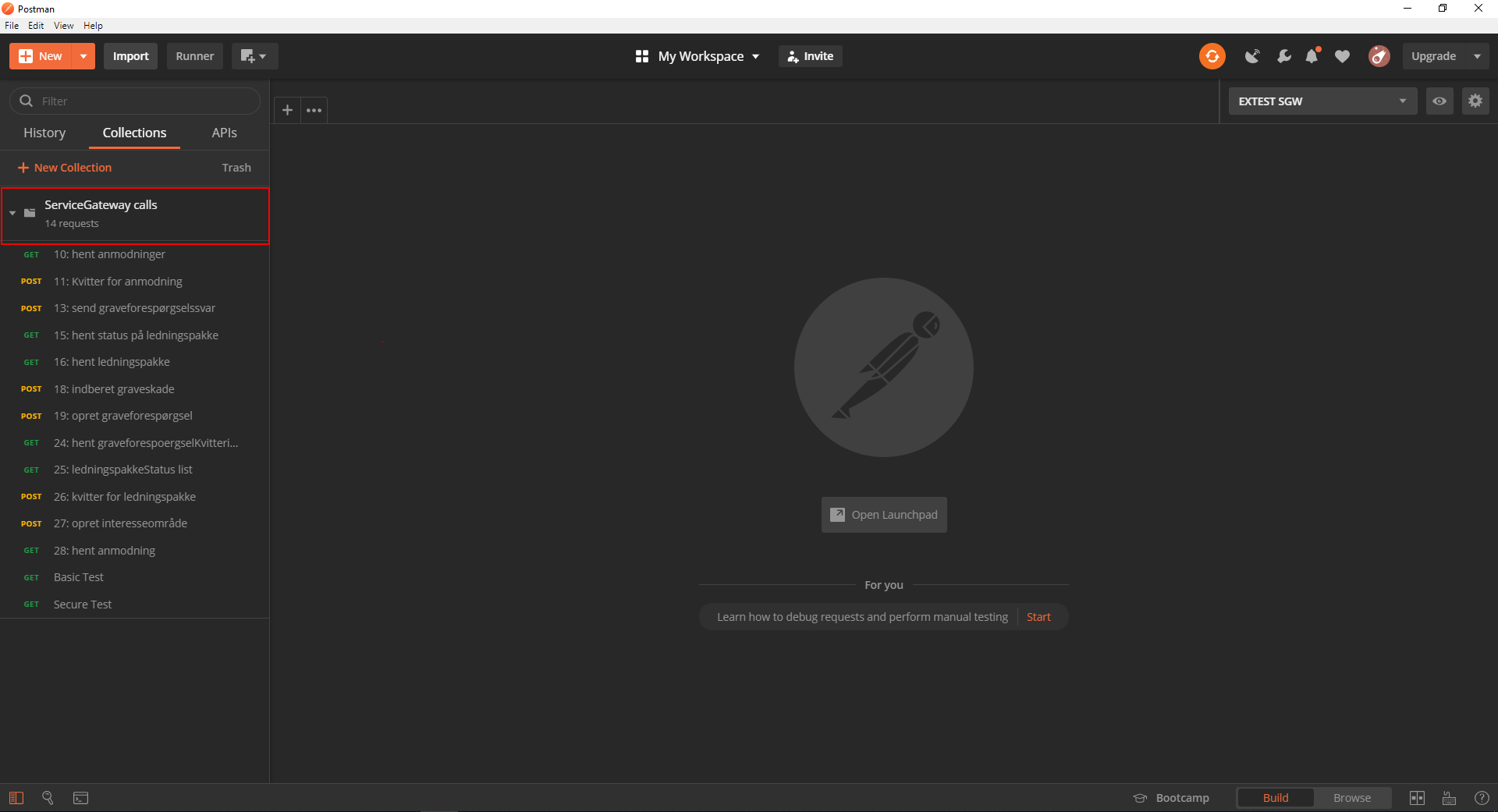
Figure 3

Figure 2 The list of possible calls to LER.

To test whether or not you can connect to the external test environment, locate either BasicTest or SecureTest at the bottom of the collection. Access by clicking either of them.

An example of a call to LER’s SecureTest targeting LER’s Extest environment is seen in Figure 3. Below the image is an explanation of how to access different environments.

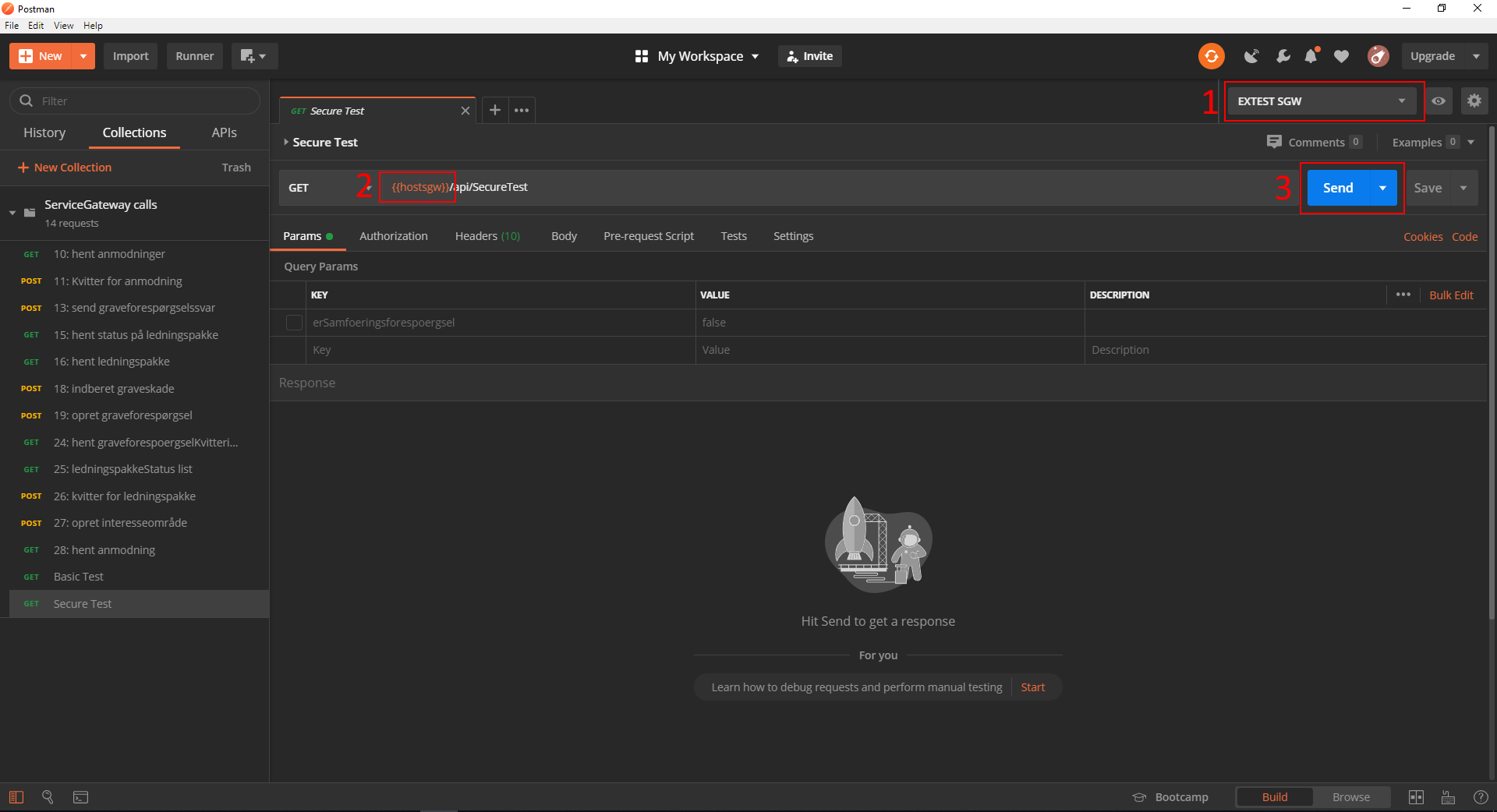


Figure 3 Using Postman to call LER's SecureTest endpoint.

The following guide explains some of the relevant information when performing tests of LER’s REST services. The number refers to the markings highlighted in Figure 3.

1. List of environments.
   1. Open to select EXTEST if it is not the default.
2. The host of the call.
3. The URL of the call.
4. Click “Send” to send a call to LER’s REST services. As the above image states: “Hit Send to get a response”. This is where the response will show.

If the call to SecureTest is successful (http status 200), all the other REST services are expected to work as well.

# Example

The following demonstration shows to use Postman as ledningsejer going through the intended flow. First “Hent Anmodninger (integration 10)” is used to find the new anmodninger, then “Kvitter for Anmodning (integration 11)” is used to get a receipt, and finally” Send graveforespørgselssvar (integration 13)” is used to send in ledningsoplysninger.

Note that the values displayed in these examples are purely for displaying purposes only, and the user should provide their own valid data to insert.

## Using Hent Anmodninger (integration 10)

Hent anmodninger is simple to use, as it will get a list of rykkere and anmodninger. Only 50 anmodninger will be shown at any time, and they are sorted by oldest to newest date.

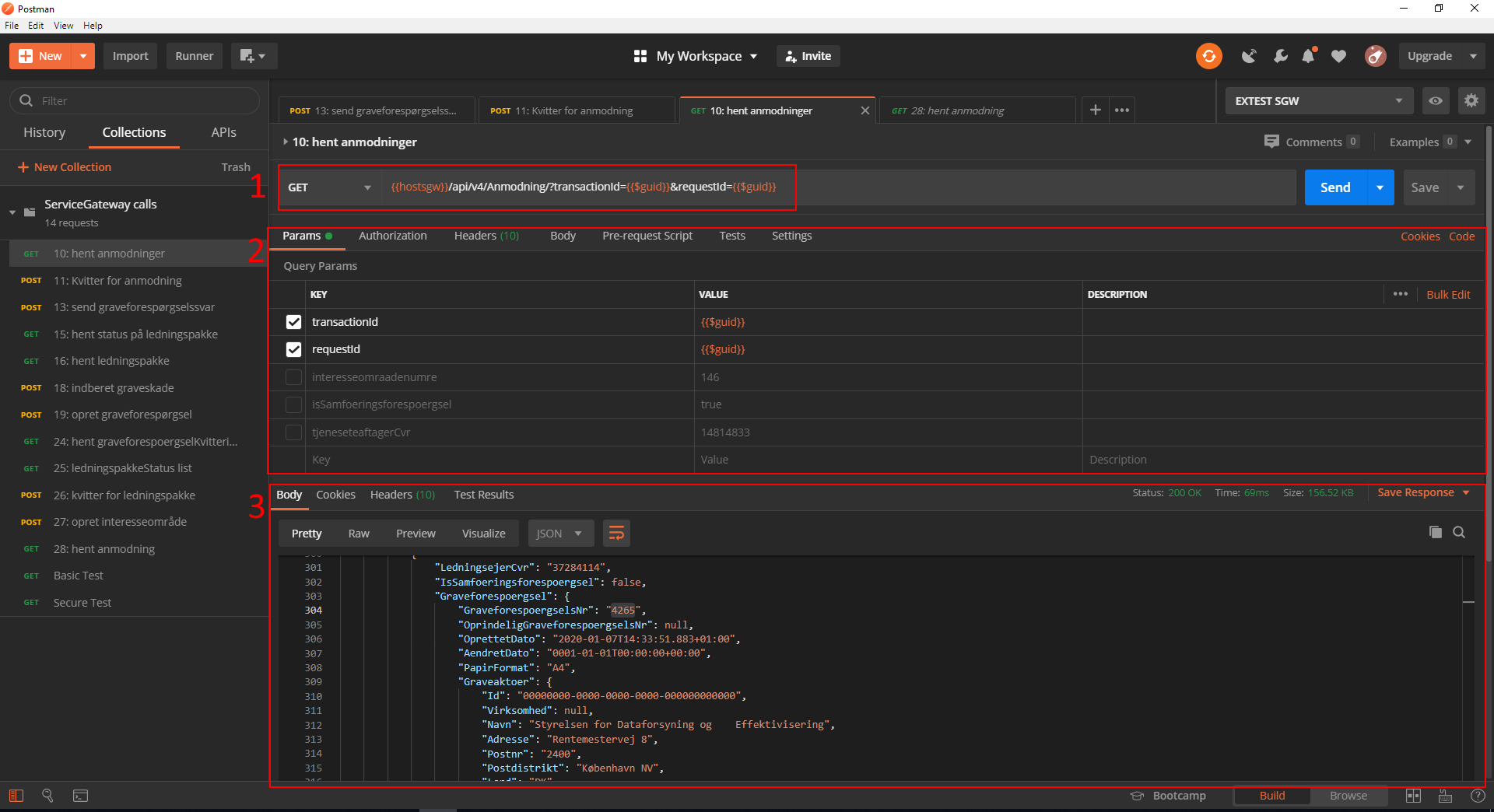


Figure 4 An example of using “Hent Anmodninger (integration 10)”.

The number refers to the markings highlighted in Figure 4.

1. We use GET as the HTTP method, and the URL contains two parameters, matching the key-value pairs in step 2.
2. The key-value pairs are defined here and can be toggled on or off by clicking the checkbox, which will automatically remove or add them from the URL.
3. The result of the query shows up here after clicking SEND. By looking at this list we see the list of anmodninger we need to consider. In the result is an anmodning for graveforespørgsel with LER-nummer 4265. The anmodning relates to interessområde with indberetningsnr 38, and this combination of LER-nummer and indberetningsnr will be used in section 3.2.

## Using Kvitter for Anmodning (integration 11)

The next step is to use “Kvitter for Anmodning (integration 11)” to get a receipt using the same LER-nummer and indberetningsnr values we got from the anmodning. This marks the anmodning as handled by the user, and it will no longer show up when “Hent Anmodninger (integration 10)” is called. Should the user require to see the anmodning, it can be fetched using “Hent Anmodning (integration 28)”.

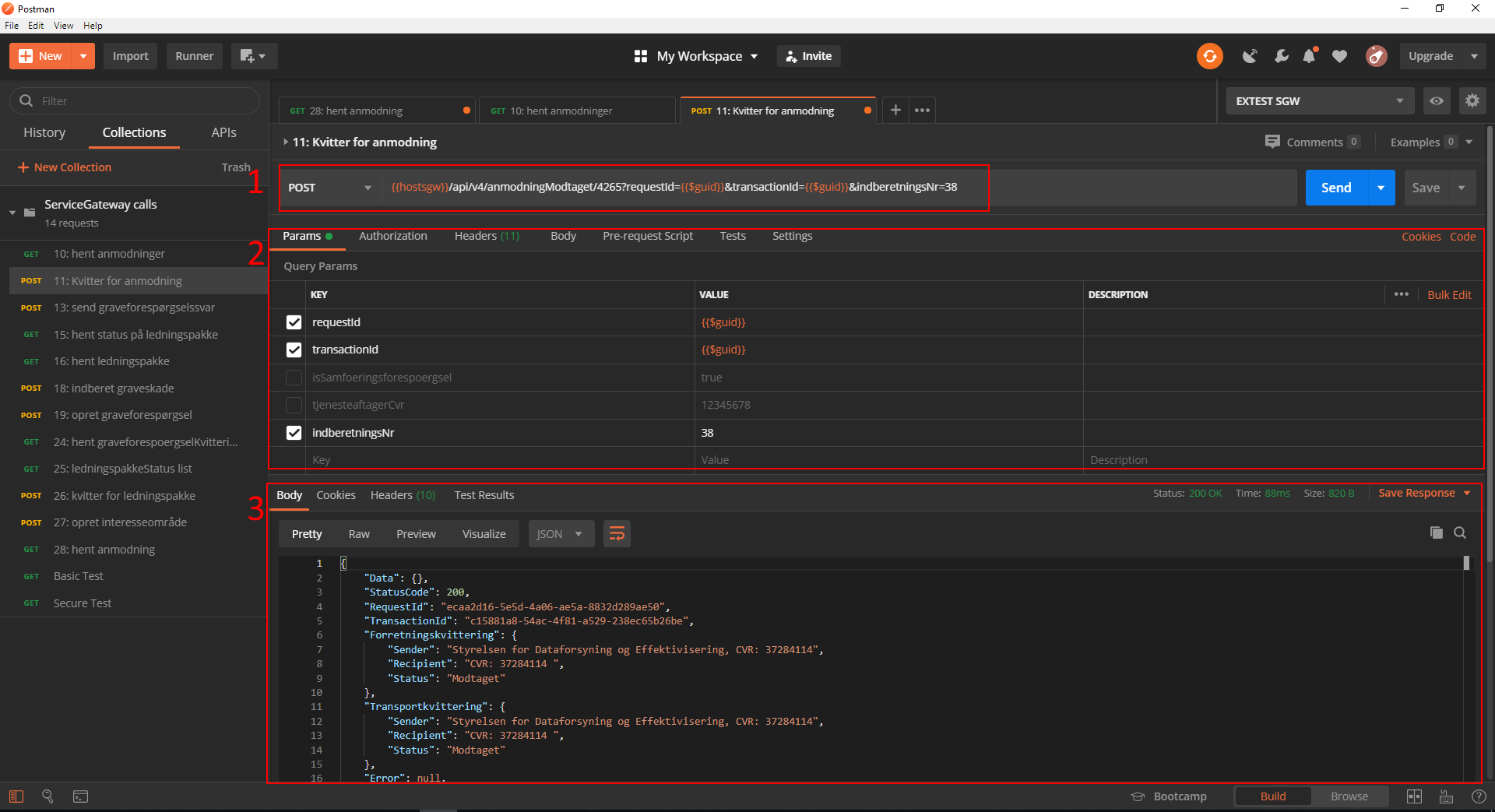


Figure 5 An example of using “Kvitter for Anmodning (integration 11)”.

The number refers to the markings highlighted in Figure 5.

1. We use POST as the HTTP method. The Id is the LER-nummer, which for this purpose is 4265.
2. We use the IndberetningsNr=38.
3. After pressing send we see that the call is successful.

## Using Send Graveforespørgselssvar (integration 13)

The final step is to send the graveforespørgselssvar. This is done by converting the GML file to base64, then adding it to the query and pressing SEND. For a guide on how to create GML see [C0200 – Udfyldelse af GML].

The first step is to convert the zipped ledningsoplysninger to base64, this can be done using <https://base64.guru/converter/encode/file>. An example of how this is done can be seen in Figure 6.

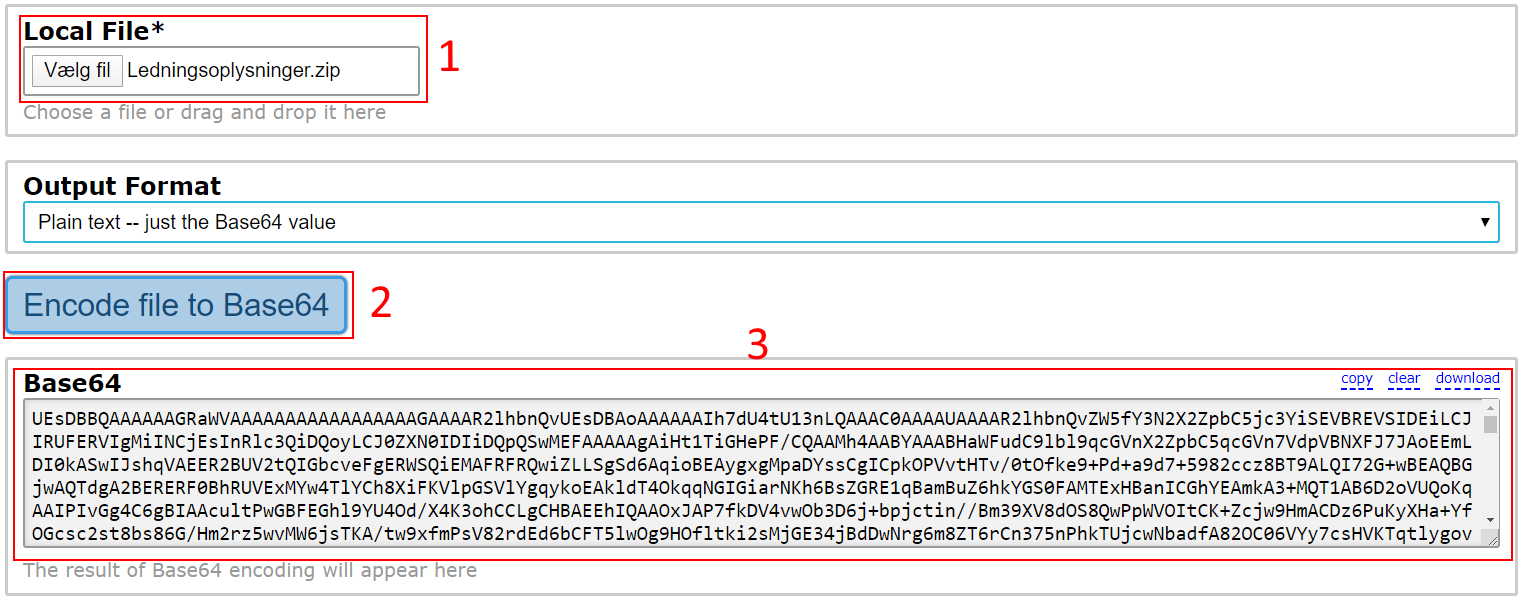


Figure 6 An example of how to convert ledningsoplysninger to base64

Now that we have the ledningsoplysninger in base64 we can set up “Send Graveforespørgselssvar (integration 13)”. This is mostly done, as is in the previous integration.

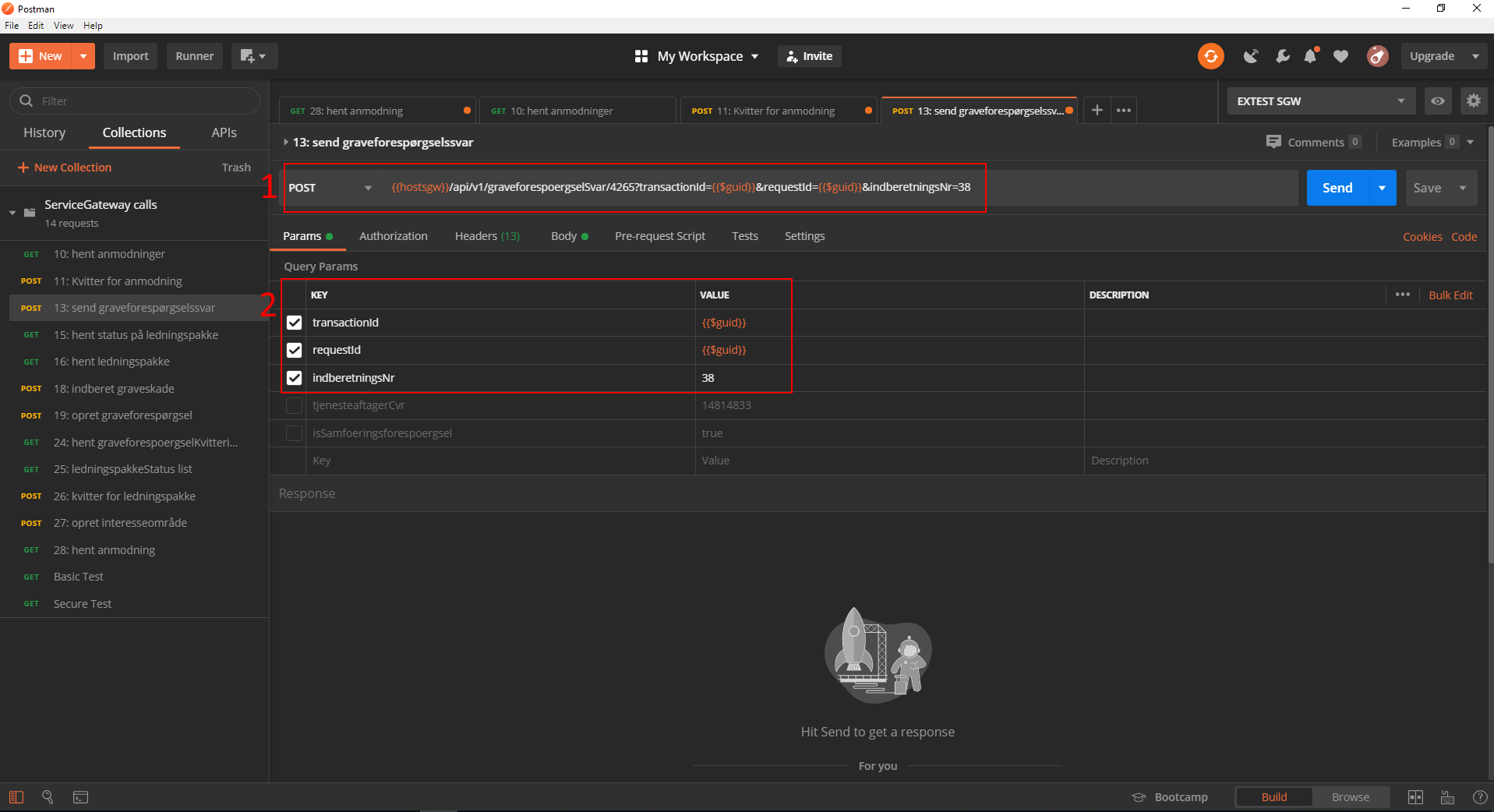


Figure 7 the parameter setup for “Send Graveforespørgselssvar (integration 13)”.

1. We use the LER-nummer 4265 as Id.
2. We set indberetningsNr=38.

After the parameters have been set up correctly, the next step is to verify that the content is sent as JSON, this can be seen in Figure 8, where we change to the “Headers” tab.

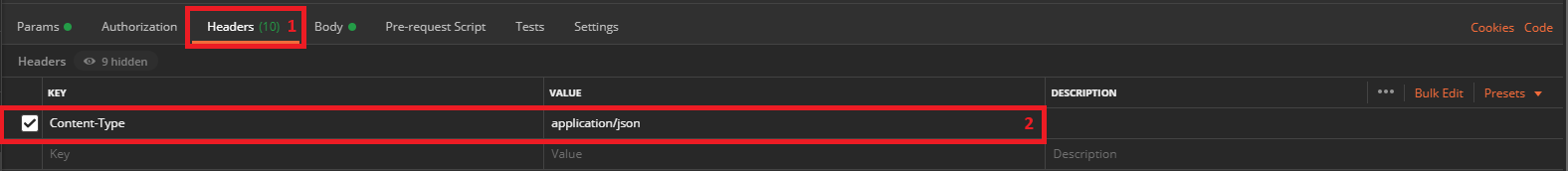


Figure 8 The header setup for “Send Graveforespørgselssvar (integration 13)”.

1. Change the tab to “Headers”.
2. Set the “Content-Type” to “application/json”.

Finally, we copy-paste the base64 data into the body, as shown in Figure 9.

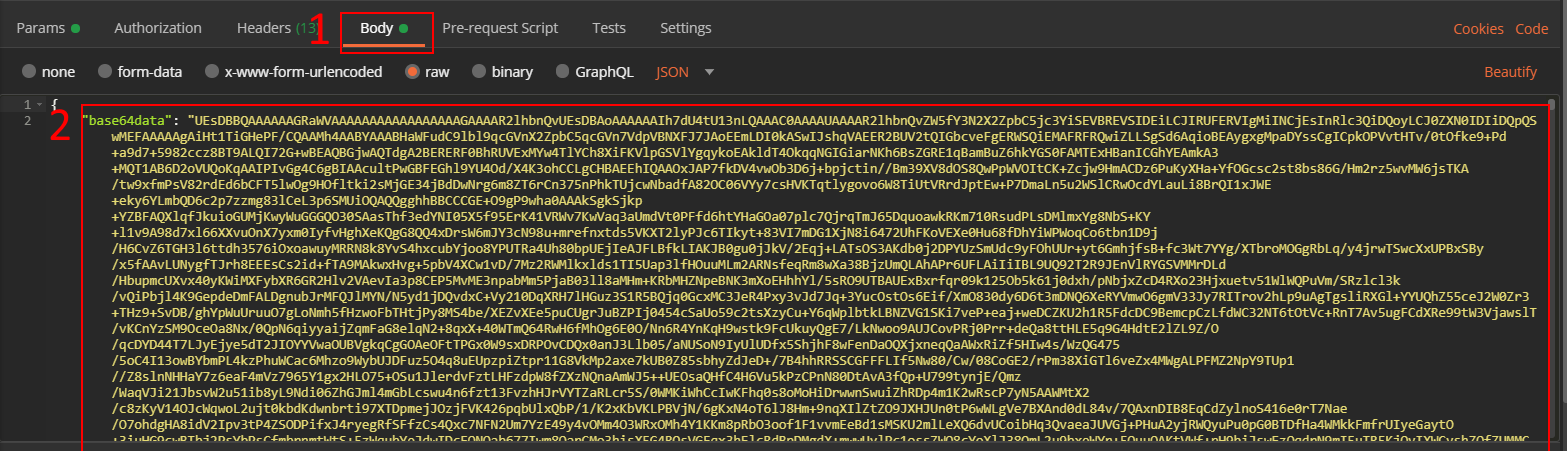


Figure 9 The body setup for “Send Graveforespørgselssvar (integration 13)”.

1. Change the tab to “Body”.
2. Insert the base64 data after the “base64data” tag, and verify that it is encapsulated by curly brackets.

We can then press “Send” and observe that the ledningsoplysninger are validated as seen in Figure 10.

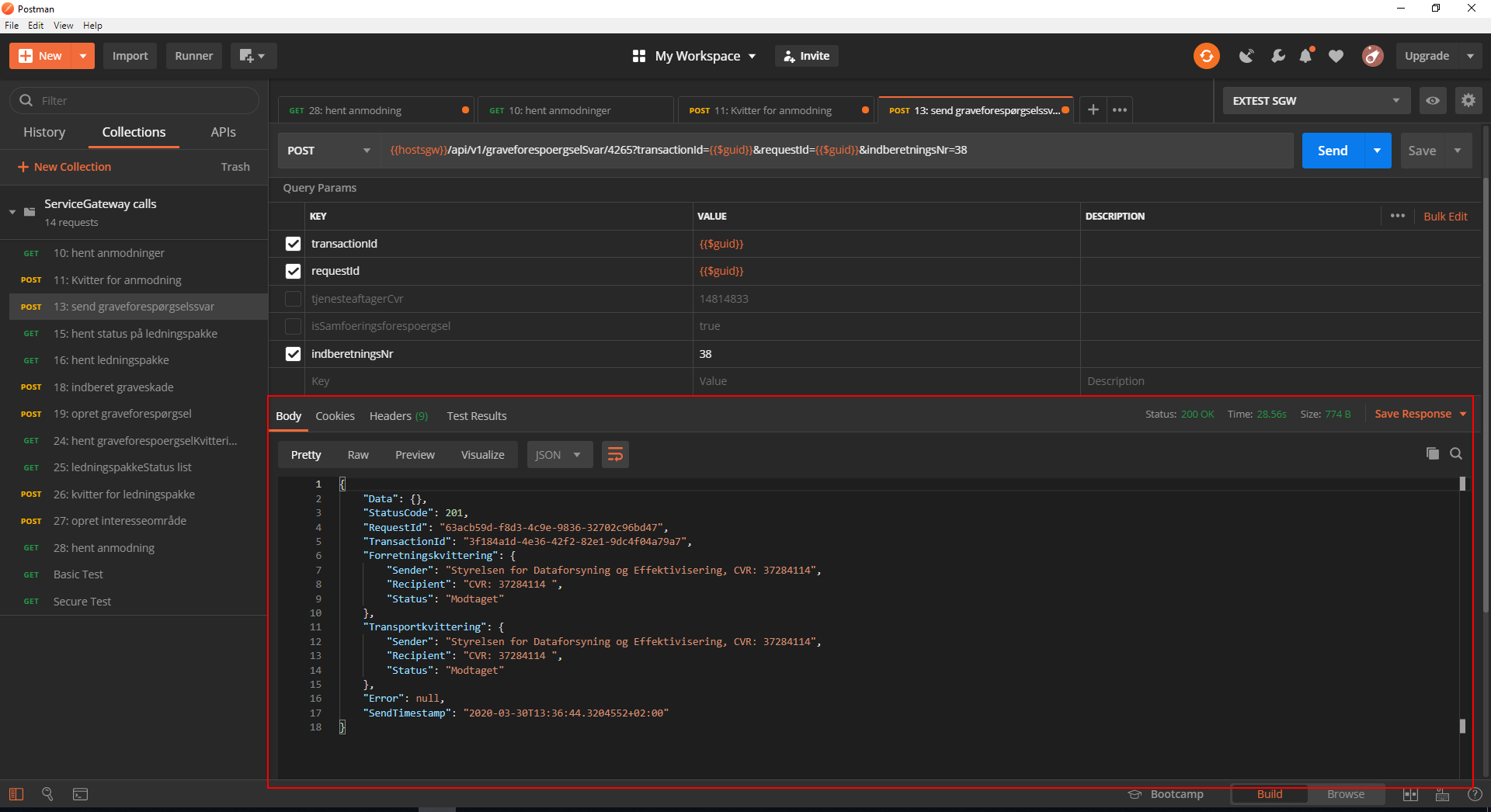


Figure 10 The ledningsoplysninger are validated.