



A direct link to your customers

– by the number one provider  
of mobile communication solutions

## LINK Mobility

# Address Lookup REST API Description

Address Lookup

Version 1.2; Last updated August 28, 2020

For help, contact [support@linkmobility.com](mailto:support@linkmobility.com)

The most up-to-date version of this document is available at

<http://www.linkmobility.com/developers/>

## Contents

Before you begin.....	3
Scope of this document .....	3
Capabilities of “Common” platform .....	3
IP addresses .....	3
Character encoding.....	3
Data types .....	3
Address .....	3
Request data (asynchronous requests only) .....	4
Response data (asynchronous requests only) .....	4
Callback address (asynchronous requests only) .....	4
Address collection (query based lookups only) .....	5
Error response.....	5
Looking up an address .....	5
Base URL.....	5
HTTP methods, statuses and actions .....	5
Basic authentication .....	5
Synchronous phone number to address.....	5
Asynchronous phone number to address.....	6
Synchronous social security number to address .....	6
Asynchronous social security number to address .....	6
Synchronous query to address .....	6
Availability.....	7
Result codes .....	7
Examples .....	8
Synchronous phone number to address.....	8
Asynchronous social security number to address .....	8
Synchronous query to address .....	10
Appendix 1 .....	11
Supported TLS versions.....	11
Changelog of this document .....	12

## Before you begin

Please make sure that Link Mobility Support has provided you with **Username**, **Password**, **PlatformId**, **PartnerId** and – if you intend to use the service asynchronously – a **GateId**. To provide a GateId support will need a result receiver URL to which addresses can be delivered.

Please make an opening in your firewall if necessary so that Common can connect to your system. For a list of the addresses Common will connect from, see below.

## Scope of this document

This document will describe the Application Programming Interface (API) to perform address lookups based on phone number (MSISDN), social security number (SSN) or queries.

Common is a REST API. Responses will be delivered in JSON format. A basic familiarity with REST APIs and JSON is assumed.

## Capabilities of “Common” platform

Common is a high-capacity, high-stability service which will allow you to make synchronous and asynchronous address lookups based on name, street, city, phone number or social security number.

## IP addresses

When delivering an address to you asynchronously, the requests will be coming from the following IP addresses:

Hostname	IP address
socks1.sp247.net	195.84.162.34
socks2.sp247.net	194.71.165.71
socks3.sp247.net	195.84.162.16
socks4.sp247.net	194.71.165.98
socks5.sp247.net	195.84.162.3
socks6.sp247.net	194.71.165.122

Please add openings in your firewalls so that these six hosts can connect to you.

## Character encoding

All communication to and from Common will be encoded using **UTF-8**.

## Data types

### Address

When requesting an address, the response will contain the following data.

Parameter	Data type	Description
firstname	String	First name of person.
middlename	String	Middle name of person.
lastname	String	Last name of person.
company	String	Company where person is employed.
address	String	The street address.
streetnumber	String	The street number.

Parameter	Data type	Description
<b>postalCode</b>	String	Postal code or zip code.
<b>area</b>	String	Area, city or locality.
<b>coAddress</b>	String	The C/O address.
<b>country</b>	String	Two letter country code (ISO3166).
<b>socialSecurityNumber</b>	String	The social security number or personal number. Country specific.
<b>birthdate</b>	String	Birthdate on the form YYYY-MM-DD
<b>mobileNumber</b>	String	Primary mobile telephone number in MSISDN format. Example: +46732594083
<b>telephoneNumber</b>	String	Primary land line telephone number in MSISDN format.
<b>customParameters</b>	KeyValue	Key-value map containing lookup provider specific data (such as sex, exact position, secondary phone numbers, etc.)

#### Request data (asynchronous requests only)

Data POSTed to service when requesting an asynchronous address lookup.

Parameter	Data type	Description
<b>refId</b>	String	This ID can be used to identify a callback address. Generated by service if not set.
<b>gateIds</b>	List<String>	A list of GateIds to which the callback address should be delivered.

#### Response data (asynchronous requests only)

Object returned by service when requesting an asynchronous address lookup.

Parameter	Data type	Description
<b>refId</b>	String	ID created by the user or the service.

#### Callback address (asynchronous requests only)

This data will be delivered when the service is used asynchronously.

Parameter	Data type	Description
<b>refId</b>	String	The ID either set in your asynchronous request or generated by the service.
<b>status</b>	Integer	0 = Success 1 = Error 2 = Address not found
<b>address</b>	Address	See above.

### Address collection (query based lookups only)

This is the object returned on query based requests.

Parameter	Data type	Description
<b>totalHits</b>	Integer	Total number of hits. If this parameter equals the size of the hits list, this collection is complete.
<b>startIndex</b>	Integer	Index of first hit in collection.
<b>hits</b>	List<Address>	See above.

### Error response

Object describing an error.

Parameter	Data type	Description
<b>status</b>	Integer	See Status codes.
<b>description</b>	String	Textual description of the problem.
<b>translatedDescription</b>	String	Textual description of the problem based on "Accept-Language".

### Looking up an address

Base URL

**https://wsx.sp247.net/addresslookup**

### HTTP methods, statuses and actions

Method	Request ok	Invalid request	Invalid login	No access	No resource
<b>GET</b>	<b>200 OK</b> Returns Address	<b>400 Bad Request</b> Returns Error Response	<b>401 Un-authorized</b> Returns Error Response	<b>403 Forbidden</b> Returns Error Response	<b>404 Not found</b> Returns Error Response
<b>POST</b>	<b>200 OK</b> Returns Request Data	<b>400 Bad Request</b> Returns Error Response	<b>401 Unauthorized</b> Returns Error Response	<b>403 Forbidden</b> Returns Error Response	-

### Basic authentication

For all requests; set the header "Authorization" to "Basic [encoded string]" where [encoded string] is a base 64 encoded string of your username and password separated by a semi colon.

### Synchronous phone number to address

GET /platform/[platformId]/partner/[partnerId]/msisdn/[msisdn]

URI Parameter	Data type	Description
<b>platformId</b>	String	Provided by Link Mobility.
<b>partnerId</b>	String	Provided by Link Mobility.
<b>msisdn</b>	String	The MSISDN to look up (exclude + sign or URL encode)

Returns an Address object or Error response on failure.

#### Asynchronous phone number to address

POST /platform/[platformId]/partner/[partnerId]/msisdn/[msisdn]

URI Parameter	Data type	Description
<b>platformId</b>	String	Provided by Link Mobility.
<b>partnerId</b>	String	Provided by Link Mobility.
<b>msisdn</b>	String	The MSISDN to look up (exclude + sign or URL encode)

A Request data object must be sent as content. Returns a Response data object. Resulting Callback address will be forwarded to provided gate(s).

#### Synchronous social security number to address

GET /platform/[platformId]/partner/[partnerId]/ssn/[ssn]

URI Parameter	Data type	Description
<b>platformId</b>	String	Provided by Link Mobility.
<b>partnerId</b>	String	Provided by Link Mobility.
<b>ssn</b>	String	The social security number to look up.

Returns an Address object or Error response on failure.

#### Asynchronous social security number to address

POST /platform/[platformId]/partner/[partnerId]/ssn/[ssn]

URI Parameter	Data type	Description
<b>platformId</b>	String	Provided by Link Mobility.
<b>partnerId</b>	String	Provided by Link Mobility.
<b>ssn</b>	String	The social security number to look up.

A Request data object must be sent as content. Returns a Response data object. Resulting Callback address will be forwarded to provided gate(s).

#### Synchronous query to address

GET /platform/[platformId]/partner/[partnerId]/query?{key}=[val]&...

URI Parameter	Data type	Description
<b>platformId</b>	String	Provided by Link Mobility.
<b>partnerId</b>	String	Provided by Link Mobility.

Query parameter (key)	Data type	Description
<b>name</b>	String	Name of person to look up. (Optional)
<b>address</b>	String	Street address of person to look up. (Optional)
<b>postalCode</b>	String	Postal code or zip code of person to look up. (Optional)
<b>area</b>	String	Area or locality. (Optional)
<b>country</b>	String	Two letter country code (ISO3166).
<b>start</b>	Integer	Index of first hit to be returned. (Optional, default = 0)
<b>results</b>	Integer	Number of hits to return (Optional, default = 20)

Returns an Address object or Error response on failure.

### Availability

Country	MSISDN to address	MSISDN to address + SSN	SSN to address	Query lookup
<b>Norway</b>	Yes	No	No	Yes
<b>Sweden</b>	Yes	Yes	Yes	Yes
<b>Denmark</b>	Yes	No	No	Yes
<b>Finland</b>	Yes	No	No	No

### Result codes

Result code	Description
<b>105000</b>	Unknown error
<b>105001</b>	Success
<b>105100</b>	Invalid authentication
<b>105101</b>	Forbidden
<b>105201</b>	Invalid PlatformId
<b>105202</b>	Invalid PartnerId
<b>105203</b>	Invalid MSISDN
<b>105204</b>	Invalid social security number
<b>105205</b>	No gates provided
<b>105300</b>	Address not found
<b>105301</b>	Lookup not supported for country
<b>105302</b>	Lookup provider error
<b>105303</b>	Configuration error

## Examples

### Synchronous phone number to address

Running the following curl command...

```
curl --user yourUsername:yourPassword \  
https://wsx.sp247.net/addresslookup/platform/yourPlatformId/partner/  
yourPartnerId/msisdn/46735555555
```

...will, assuming the subscriber exists, give the following Address object as response:

```
{  
  "firstname": "Test",  
  "middlename": null,  
  "lastname": "Testson",  
  "company": null,  
  "address": "Testroad",  
  "streetnumber": "2",  
  "postalCode": "12345",  
  "area": "Test town",  
  "coAddress": null,  
  "country": "SE",  
  "socialSecurityNumber": null,  
  "birthdate": "1970-01-01",  
  "telephoneNumber": null,  
  "mobileNumber": "+46735555555",  
  "customParameters": {  
    "apartment": "lgh 1002",  
    "sex": "M"  
  }  
}
```

If the subscriber is non-existent, the following Error response will be returned:

```
{  
  "status": 105300,  
  "description": "Address not found"  
}
```

### Asynchronous social security number to address

Running the following curl command...

```
curl -XPOST \  
-H "Content-Type: application/json" \  
-d "{\"refId\":\"ref1\", \"gateIds\":[\"gate1\"]}" \  
-u yourUsername:yourPassword \  

```



```
https://wsx.sp247.net/addresslookup/platform/yourPlatformId/partner/yourPartnerId/ssn/7001010000
```

...will return the following Response data object:

```
{
  "refId": "ref1"
}
```

If the subscriber exists, the following Callback address will be forwarded to the URL(s) associated with gate1:

```
{
  "requestStatus": 0,
  "refId": "ref1",
  "address": {
    "firstname": "Test",
    "middlename": null,
    "lastname": "Testsson",
    "company": null,
    "address": "Test street",
    "streetnumber": "2",
    "postalCode": "12622",
    "area": "Test area",
    "coAddress": null,
    "country": "SE",
    "socialSecurityNumber": "00000000",
    "birthdate": "1981-05-05",
    "telephoneNumber": null,
    "mobileNumber": "+46730000000",
    "customParameters": {
      "apartment": "lgh 1002",
      "sex": "M"
    }
  }
}
```

If the subscriber does not exist, the resulting callback address object will look like this:

```
{
  "requestStatus": 2,
  "refId": "ref1",
  "address": null
}
```

### Synchronous query to address

Running the following curl command...

```
curl -XGET \  
-u yourUsername:yourPassword \  
https://wsx.sp247.net/addresslookup/platform/SEND/partner/1/query?name=Test+Testsson&area=Test+town&country=SE
```

...will, assuming any subscribers exists, give the following Address collection object as response:

```
{  
  "totalHits": 1,  
  "startIndex": 0,  
  "hits": [  
    {  
      "firstname": "Test",  
      "middlename": null,  
      "lastname": "Testsson",  
      "company": null,  
      "address": "Test street",  
      "streetnumber": "76",  
      "postalCode": "12345",  
      "area": "Test town",  
      "coAddress": null,  
      "country": "SE",  
      "socialSecurityNumber": null,  
      "birthdate": "1970-01-01",  
      "telephoneNumber": null,  
      "mobileNumber": "+46755555555",  
      "customParameters": {  
        "latitude": "37.397914",  
        "longitude": "-116.227283"  
      }  
    }  
  ]  
}
```

## Appendix 1

### Supported TLS versions

From 2020-11-15 will TLS 1.2 or higher be required for all HTTPS connections.

Support for TLS 1.0 and 1.1 will be discontinued. Versions 1.0 and 1.1 of TLS are older protocols that have been deprecated and are considered as security risks in the Internet community.

LINK strongly recommend using HTTPS if HTTP is being used today. HTTP is deprecated as of 2020-09-01 by LINK and will be removed in the future. Date for HTTP removal is not yet decided.

## Changelog of this document

<b>Date</b>	<b>Version</b>	<b>Author</b>	<b>Changes</b>
<b>2015-12-07</b>	1.0	JA	Initial version
<b>2017-08-24</b>	1.1	KCN	Minor changes
<b>2020-08-28</b>	1.2	EPT	Appendix 1 added