LBA Sense - API Documentation

Release date: 26/04/2023	**Version:** 12.1.0
Change Log: UniqueVisitorCount, RSSIs and RSSISetUp APIs added	

API Overview

LBASense provides many APIs that enable you to harness the power of LBASense solutions.

This page describes the APIs that are available on the platform, and how to use them.

Which APIs are available ?

The LBASense developer tools and endpoints are grouped into the following APIs.

API name	Description
VisitorCount	This API lets you query your historical data by defining a time frame, resolution and more.
FloatingPopulation	This API lets you query your realtime data.
SensorHealth	This API lets you query the current state of your sensor (online/offline, software version,etc.)
Tracks	This API enables you to download the track files (i.e: raw data) in csv format.

How do I get started ?

Basic Requirements

All APIs described above require the user to provide their user credentials (username + password) to authenticate the request. Please get in touch with the LBA Team if you do not already possess user credentials.

Concepts

Parameter	Explanation
site	site is the logical collection of sensors where all data collected is processed and fused into a single Situation Awareness Picture. Usually, a site includes multiple regions
region	region is the logical area with geographic boundaries and some logical similarities. Regions usually reflect the detection ranges of the sensors; they extend according to customers' requirements
sensor	sensor is the physical device, uniquely identified by a numerical ID assigned by the LBASense system, that detects anonymous signals

How to get started with LBASense APIs in Local Networks server

When 3rd party users utilize LBASense APIs in Local Networks, the **URL Base Path** may differ but the behavior of APIs is as depicted in the description file.

Consult with the LBASense Team to get the URL Base Path applicable to you

Additionally, the User may encounter SSL certificate-related errors on *Local Network Servers*, in such case, the web-application should disable or bypass these errors.

Developer documentation

This documentation will detail the following:

Querying Parameters

LBASense APIs basic querying parameters are shared among all web-services (unless otherwise indicated). Other call-related parameters exist and are described in the relevant API section. Response parameters depend on the query type.

Mandatory Parameters

Name	Value	Required	Summary
user	String	Yes	The username you use to access the LBASense dashboard
pass	Integer	Yes	The password you use to access the LBASense dashboard
site_id	String	Yes	The unique ID of the site for which the query will retrieve the data
domain	String	Yes	The domain given to you by the LBA administration

API-specific parameters will be described for each API.

API Success and Error Responses

Success Responses

LBASense APIs return a 200 HTTP status code if a request is successful, in such a case the resource has been fetched and is transmitted in the message body.

The returned message body may contain data for non-existing or incomplete data sets, in such case the relevant properties of the response body are counter-signed with value -1.

Example for the properties of the response body structure for non-existing or incomplete data set:

```
{
  "data": [
    {
       "date": "2023-03-21",
       "region_data": [
           {
              "region": 1,
              "count": -1
           }
         ]
       }
    ]
}
```

Error Responses

If an error occurs during the processing of a query, the API returns an error code and the reason is described in the response.

The third-party application that uses LBASense APIs needs to support proper error handling.

Most occurring error codes are summarized below, others exist and may occur.

- *401* for Unauthorized (i.e. invalid credentials);
- *403* for Forbidden (i.e. invalid permission);
- *404* for Bad Request;
- *500* for Internal Server Error.

VisitorCount API

This call returns the **total number of visitors** for the entire site and its various regions. The web-service and the other query parameters are defined below.

Method Type	**GET**
Webservice End Point	https://{domain}/api/Analytics/VisitorCount


In most cases, {domain} value is `master.lbasense.com`

Mandatory querying parameters

Name	Value	Required	Summary
user	String	Yes	The username you use to access the LBASense dashboard
pass	Integer	Yes	The password you use to access the LBASense dashboard
site_id	Integer	Yes	The unique ID of the site for which the query will retrieve the data
start_time	String	Yes	Begin time of the period for the query. Supported format is `YYYY-MM-DDTHH:mm:ss` (ISO 8601 format) *Note*: All the times must be specified according to the site time zone.
end_time	String	Yes	End time of the period for the query. Supported format is `YYYY-MM-DDTHH:mm:ss` (ISO 8601 format) *Note*: All the times must be specified according to the site time zone.
resolution	String	Yes	It is defined as `days`, `hours` and it determines the format of results that the query will return for the requested period.

Sample Call

```
https://{domain}/api/Analytics/VisitorCount/?format=json&site_id={site_id}&user={username}&pass={password}&start_time={start_time}&end_time={end_time}&resolution={resolution}
```

Sample Response


```
{
  "data": [
    {
      "date": "2023-03-21",
      "region_data": [
          {
             "region": 0,
             "count": 478
          },
         {
             "region": 1,
             "count": 478
         }
       ]
    }
  ]
}
```


Region 0 is the aggregation of all regions in the site. When there is only one region in a site, R0 = R1

UniqueVisitorCount API

This call returns the **unique number of visitors** for the entire site and its various regions on a daily basis only. The web-service and the other query parameters are defined below.

Method Type	**GET**
Webservice End Point	https://{domain}/api/Analytics/UniqueVisitorCount

In most cases, {domain} value is `master.lbasense.com`

Mandatory querying parameters

Name	Value	Required	Summary
user	String	Yes	The username you use to access the LBASense dashboard
pass	Integer	Yes	The password you use to access the LBASense dashboard
site_id	Integer	Yes	The unique ID of the site for which the query will retrieve the data
start_time	String	Yes	Begin time of the period for the query. Supported format is `YYYY-MM-DDTHH:mm:ss` (ISO 8601 format) *Note*: All the times must be specified according to the site time zone.
end_time	String	Yes	End time of the period for the query. Supported format is `YYYY-MM-DDTHH:mm:ss` (ISO 8601 format) *Note*: All the times must be specified according to the site time zone.

Sample Call

```
https://master.lbasense.com/api/Analytics/UniqueVisitorCount/?site_id={site_id}&start_time={start_time}&end_time={end_time}&user={username}&pass={password}
```

Sample Response


```
{
  "data": [
    {
      "date": "2023-03-21",
      "region_data": [
           {
              "region": 0,
              "count": 478
           },
          {
              "region": 1,
              "count": 478
          }
       ]
    }
  ]
}
```


Region 0 is the aggregation of all regions in the site. When there is only one region in a site, R0 = R1

Response fields

Property Name	Type	Description
	object	Main object with 1 property
data	array	Array of data objects with 2 properties
data.date	string	Statistics relevance date (or date hour)
data.region_data[ ]	array	Array of visitors count data for site's regions, or queried region if the filter is applied.
region_data[ ] .region	integer	Indicate the ID of the region to which statistics apply to.
region_data[ ].count	integer	The total number of unique visitors for the date, time and location.

FloatingPopulation API

This call returns the real time Population Count for the site's regions. The web-service and the other query parameters are defined below.

Method Type	**GET**
Webservice End Point	https://{domain}/api/RealTime/FloatingPopulation

In most cases, {domain} is `master.lbasense.com`

Mandatory querying parameters

Name	Value	Required	Summary
user	String	Yes	The username you use to access the LBASense dashboard
pass	Integer	Yes	The password you use to access the LBASense dashboard
site_id	Integer	Yes	The unique ID of the site for which the query will retrieve the data

Sample Call

```
https://{domain}/api/RealTime/FloatingPopulation?user={username}r&pass={password}&site_id={site_id}
```

Sample Response

```
[
   {
      "region": 0,
      "numVisitors": 142
    }
    {
       "region": 1,
       "numVisitors": 142
    }
 ]
[
```

Region 0 is the aggregation of all regions in the site. When there is only one region in a site, R0 = R1

Response fields

Property Name	Type	Description
	array	Main array. The array contains as many objects as the number of regions and region 0 for the queried site.
	object	Real time statistics for the indicated region.
region	integer	ID of region to which statistics apply to.
numVisitors	integer	Unique count of visitors.

SensorHealth API

This call returns the current status of LBASense sensors for a queried site. The web-service and the other query parameters are defined below.

Method Type	**GET**
Webservice End Point	https://{domain}/api/Hub/Management/SensorHealth/

In most cases, {domain} is `master.lbasense.com`

Mandatory querying parameters

Name	Value	Required	Summary
user	String	Yes	The username you use to access the LBASense dashboard
pass	Integer	Yes	The password you use to access the LBASense dashboard
site_id	Integer	Yes	The unique ID of the site for which the query will retrieve the data

Sample Call

```
https://{domain}/api/Hub/Management/SensorHealth/?user={usernam
e}pf&pass={password}&site_id={site_id}
```

Sample Response

```
[
   {
      "sensorId": 123,
      "lastTimeSeen": "YYYY-MM-DDTHH:MM:SS",
      "region": 1,
      "uptime": "HH:MM:SS",
      "status": "online",
      "softwareVersion": "1.0.0",
      "hardwareType": "unisem"
    }
 ]
```

Response fields

Property Name	Type	Description
	array	Main array. The array contains as many objects as the number of sensors injecting data in the queried site.
	object	Status and details for the indicated sensor.
sensorId	integer	Unique ID of sensor to which details apply to.
lastTimeSeen	string	The last time the sensor reported to LBASense backend, defined with site timezone.
region	integer	ID of region to which the data applies to.
uptime	string	It indicates the time since the sensor was given electricity and is defined in the format of hh:mm:ss. If the power is lost, then the sensor loses power and the uptime count starts.
status	string	Is defined as "online" if the sensor has transmitted in the last 10 minutes from the query time, if not is shown as "offline".
softwareVersion	string	Sensor firmware details.
hardwareType	string	Other hardware related details.

Track API

This API returns a zip file of csv files that detail the raw data collected by the sensors. The web-service and the other query parameters are defined below.

Method Type	**GET**
Webservice End Point	https://{domain}/api/Utils/DownloadTrackFile

In most cases, {domain} is `master.lbasense.com`

We recommend querying this API in your web browser directly.

Mandatory querying parameters

Name	Value	Required	Summary
user	String	Yes	The username you use to access the LBASense dashboard
pass	Integer	Yes	The password you use to access the LBASense dashboard
site_id	Integer	Yes	The unique ID of the site for which the query will retrieve the data
start_time	String	Yes	Begin time of the period for the query. Supported format is `YYYY-MM-DD` *Note*: All the times must be specified according to the site time zone.
end_time	String	Yes	End time of the period for the query. Supported format is `YYYY-MM-DD` *Note*: All the times must be specified according to the site time zone.

Sample Call

```
https://{domain}/api/Utils/DownloadTrackFile?user={username}&pass={password}&site_id={site_id}&start_date=YYYY-MM-DD&end_date=YYYY-MM-DD
```

CSV fields

Column Name	Type	Description
site_id	integer	The unique ID of the site for which the query will retrieve the data
region_id	integer	ID of region to which data applies to.
visitor_id	integer	The unique ID for a detected visitor
first_time_seen	string	The time in `YYYY-DD-MMTHH-MM-SS` that a visitor ID has been detected by a sensor for the first time
last_time_seen	string	The time in `YYYY-DD-MMTHH-MM-SS` that a visitor ID has been detected by a sensor for the last time
rssi_avg	integer	Average value of RSSI signal for a specific visitor ID. NB : RSSI values are always negative.
manufacturer	string	Name of device manufacturer. If this data is unavailable, the value is `NONE`
detection_type	string	`Detection_type` can be either `Real MAC`, `Connected to WiFi` or `Randomized MAC`.

RSSI APIs

Users can query two APIs:

- the **RSSI API** which displays the current RSSI values for sensors in a site
- the **RSSISetUp** API which updates RSSI values

IMPORTANT: some sites may not be compatible with these APIs.
Reach out to your customer representative if you are encountering errors when using them.

- **Checking current RSSI values**

Method Type	**GET**
Webservice End Point	https://{domain}/api-master/api/Sensors/RSSIs


In this case, {domain} is `dashboard.lbasense.com`

Mandatory querying parameters

Name	Value	Required	Summary
user	String	Yes	The username you use to access the LBASense dashboard
pass	Integer	Yes	The password you use to access the LBASense dashboard
site_id	Integer	Yes	The unique ID of the site for which the query will retrieve the data

Sample Call

```
https://dashboard.lbasense.com/api-master/api/Sensors/RSSIs?site_id={site_id}&user={username}&pass={password}
```

Sample Response

```
[
   {
      "region": 1,
      "sensor": 3012
      "rssi": -100,
    }
 ]
```

Response fields

Property Name	Type	Description
	array	Main array. The array contains as many objects as the number of sensors injecting data in the queried site.
	object	Status and details for a specific sensor.
region	integer	Unique ID of region to which the data applies to.
sensor	integer	Unique ID of sensor to which the data applies to.
rssi	integer	From -100 to 0 : the RSSI value (in db) for a specific sensor/region

- **Updating RSSI values**

Method Type	**GET**
Webservice End Point	https://{domain}/api-master/api/Sensors/RSSISetUp

In this case, {domain} is `dashboard.lbasense.com`

It can take up to 5 minutes to have the changes reflected at the sensor level.

Mandatory querying parameters

Name	Value	Required	Summary
user	String	Yes	The username you use to access the LBASense dashboard
pass	Integer	Yes	The password you use to access the LBASense dashboard
site_id	Integer	Yes	The unique ID of the site for which the query will apply
region_id	Integer	Yes	The unique ID of the region the update will be applied to. `Region_id` can be 0. In that case, all sensors will update to the selected RSSI value.
sensor_id	Integer (array)	Yes	The ID(s) of the sensor(s) the RSSI update will be applied to. When `region_id=0`, `sensor_id` should be empty (i.e: `sensor_id=[]`)
rssi	Integer	Yes	`RSSI` must be between -100 to 0. The closer to 0, the more restrictive the detection range is.

Sample Call

```
https://dashboard.lbasense.com/api-master/api/Sensors/RSSISetUp
?site_id={site_id}&user={username}&pass={password}&sensor_id=[{
sensor_id}]&rssi={rssi}&region_id={region_id}
```

Sample Response

```
[
   {
      "region": 1,
      "sensor": [
          {
               "Id": 3012,
               "Rssi": -100
          }
      ]
   }
]
```

Response fields

Property Name	Type	Description
	array	Main array. The array contains as many objects as the number of sensors injecting data in the queried site.
	object	Status and details for a specific region
region	integer	Unique ID of the region the update applies to
sensor	array	Status and details for a specific sensor
id	integer	Unique ID of the sensor the update applies to
rssi	integer	New RSSI value for the specific region/sensor

API documentation ends here.