



Affiliate API





Affiliate API

Protocol

Our API uses HTTP protocol as a transport. In developing API, architectural constraints for RESTful API services have been met. All API responses are in the common used JSON format. The Protocol is built for cross-platform interactions with external systems.

Relevant HTTP status codes

Status	Description
200 OK	Successful
201 Created	Created
400 Bad Request	Bad, incorrect request
401 Unauthorized	Unauthorized (not introduce)
500 Internal Server Error	Internal server error

Relevant HTTP methods

Method	Description
GET	Getting a resource
POST	Creating a resource





Relevant HTTP request headers

Header	Description
Content-Type	MIME type of request body - <i>application/json</i>
Accept	Command server to send only definite MIME types - <i>application/json</i>

Relevant HTTP response headers

Header	Description
Content-Type	MIME type of response body - <i>application/json</i>
Content-Encoding	A method for encoding contents of entity when transferring - <i>gzip</i>
Transfer-Encoding	List of code methods have been applied to transferring message - <i>chunked</i>
Connection	Data on routing - <i>keep-alive</i>





Authorisation

Any action with the Affiliate API should be made with the Token parameter generated inside your system. To generate a token, you must create one in your LD system for a static IP address/addresses.

Response 401

```
{
  "message": "401 Unauthorized"
}
```

List Conversions

Request

Path: */api/v1/affiliates/conversions*

Method: GET

Parameters:

- *token* (string, authorisation token. Required)
- *created_from* (string, beginning date. Optional)
- *created_to* (string, end date. Optional)
- *goal_type_id* (integer, Goal Type ID. Optional)
- *page* (integer, page number. Optional)
- *per_page* (integer, number of elements on the page. Optional)



Response 200

JSON fields:

- *data.id* (integer, ID of conversion)
- *data.lead_id* (integer, ID of lead)
- *data.created_at* (string, date of conversion)
- *meta.pagination* (pagination information)

```
{
  "data": [
    {
      "id": 1002,
      "lead_id": 10123,
      "goal_type_id": 3,
      "created_at": "2021-01-28T02:05:12+00:00"
    }
  ],
  "meta": {
    "pagination": {
      "total": 1,
      "count": 1,
      "per_page": 100,
      "current_page": 1,
      "total_pages": 1,
      "links": []
    }
  }
}
```

List Leads

Request

Path: `/api/v1/affiliates/leads`

Method: GET

Parameters:

- `token` (string, authorisation token. Required)
- `created_from` (string, beginning date. Optional)
- `created_to` (string, end date. Optional)
- `conversion_from` (string, conversion beginning date. Optional)
- `conversion_to` (string, conversion end date. Optional)
- `page` (string, page number. Optional)
- `per_page` (string, numbers of items on the page. Optional)
- `has_conversion` (integer, has or hasn't a conversion. Optional)
- `tp_source` (string. Optional)

Response 200

JSON fields:

- `data.id` (Lead ID)
- `data.has_conversion` (Conversion Yes/No)
- `data.conversion_date` (date of conversion)
- `data.created_at` (Lead creation date)
- `meta.pagination` (pagination information)

```
{
  "data": [
    {
      "id": 23,
      "tp_affiliate_id": "1",
      "conversion_date": null,
      "country_code": "RU",
      "tp_source": "API",
      "has_conversion": false,
      "tp_aff_sub": "",
      "tp_aff_sub2": "",
      "tp_aff_sub3": "",
      "tp_aff_sub4": "",
      "tp_aff_sub5": "",
      "created_at": "2021-01-28T00:59:48+00:00",
      "sale_status": "New"
    }
  ],
  "meta": {
    "pagination": {
      "total": 1,
      "count": 1,
      "per_page": 100,
      "current_page": 1,
      "total_pages": 1,
      "links": []
    }
  }
}
```

Lead Profile get fields

Request

Path: `/api/v1/affiliates/fields`

Method: GET

Parameters:

- `token` (string, authorisation token. Required)

Response 200

JSON fields:

- `data.key` (field key)
- `data.type` (field type. Possible types are: email, text, phone, date, password, accepted, select-countries)
- `data.label` (field label)

```
{
  "data": [
    {
      "key": "email",
      "type": "email",
      "label": "Email"
    },
    {
      "key": "phone",
      "type": "phone",
      "label": "Phone"
    }
  ]
}
```


Goal Types get

Request

Path: `/api/v1/affiliates/goal-types`

Method: GET

Parameters:

- `token` (string, authorisation token. Required)

Response 200

JSON fields:

- `data.id` (Type ID)
- `data.name` (Type name)

```
{
  "data": [
    {
      "id": 1,
      "name": "FTD"
    }
  ]
}
```

Push Lead

Request

Path: `/api/v1/affiliates/leads`

Method: POST

Parameters:

- `token` (string, authorisation token. Required)

Form parameters:

- `ip` (string, Lead IP Required)
- `tp_source` (string. Optional)
- `tp_aff_sub` (string. Optional)
- `tp_aff_sub2` (string. Optional)
- `tp_aff_sub3` (string. Optional)
- `tp_aff_sub4` (string. Optional)
- `tp_aff_sub5` (string. Optional)
- `profile[{{field.key}}]` (Lead Profile fields values. For instance - `profile[email]`, `profile[phone]`)
- `tp_id` (string. Optional, ID of traffic provider)
- `tp_offer_id` (string. Optional, ID of the offer)
- `funnel_id` (string, Optional, ID of the funnel)


Response 500

JSON fields:

- `error` (error message)

```
{
  "error": "Lead is not pushed to Advertiser"
}
```





Response 400

JSON fields:

- *validation_errors* (validation error message)

```
{
  "validation_errors": {
    "profile.email": [
      "The profile.email must be a valid email address."
    ]
  }
}
```

Response 200

JSON fields:

- *lead_id* (Lead ID)
- *auto_login_url* (auto login URL)
- *advertiser_name* (name of advertiser to which lead has been pushed)

```
{
  "lead_id": 2371,
  "auto_login_url": "https://autologindomain.com/index.php?auto\_login\_param=value",
  "advertiser_name": "Advertiser Name"
}
```

