



PROFITSHARE APPLICATION  
PROGRAMMING INTERFACE  
(FOR AFFILIATES)

## Contents

Authentication .....	2
Constructing the authentication string.....	2
Possible authentication errors.....	2
API Reference.....	3
API Methods.....	3
Advertisers .....	3
Advertisers listing.....	3
Campaigns.....	4
Campaigns listing .....	4
Links .....	5
Links add .....	5
Commissions .....	6
Commission listing .....	6
Products .....	8
Products listing.....	8
Sample requests.....	10
Getting the active advertisers list .....	10
Advertiser products .....	10
Link generator .....	11

## Authentication

### Constructing the authentication string

In order to use the API you need an API USER and an API KEY, they are both provided when signing up.

Each resource can handle one or more of the following HTTP methods : GET, PUT, DELETE, POST.

There are several required HTTP headers :

- Date - for example: Wed, 01 Feb 2008 12:00:00 GMT
- X-PS-Client - for example: test-account
- X-PS-Accept - it can take 2 values: xml or json
- X-PS-Auth - for example: 90a6d325e982f764f86a7e248edf6a660d4ee833

The value of X-PS-Auth header is used for authentication of the request. See below the way this header is constructed:

- step 1 : signature\_string = concatenate strings (HTTP verb, API Method, Query String, X-PS-Client header value, Date header value)
- step 2 : X-PS-Auth header value = HMAC-SHA1 (API KEY, signature\_string)

The HMAC-SHA1 algorithm is explained in detail here <http://en.wikipedia.org/wiki/HMAC#Implementation>

### Sample PHP X-PS-Auth Construction:

```
<?php
    $api_user = 'test-account';
    $api_key = '5f4dbf2e5629d8cc19e7d5187426667809ddb677';

    $query_string = 'date_from='.urlencode('2013-05-01').'&date_to='.urlencode('2013-05-31');
    $date = 'Wed, 01 Feb 2008 12:00:00 GMT';

    $signature_string = 'GETaffiliate-commissions/?' . $query_string . '/'.
    $api_user. $date;
    $signature_string_encrypted = hash_hmac('sha1', $signature_string, $api_key);
?>
```

### Possible authentication errors

- AuthHeaderMissing - this means that the X-PS-Auth header was not detected in your request
- AuthTimeDifference - this means that the time difference between the request time and our server time is greater than 20 seconds. Please read the important note below
- ClientHeaderMissing - this means that the X-PS-Client header was not detected in your request

- DateHeaderMissing - this means that the X-PS-Date header was not detected in your request
- InvalidClient - the supplied API USER is not registered with our system
- InvalidSignature - the supplied signature in X-PS-Auth header is wrong. Possible reasons are: the signature\_string was not properly computed, the HMAC algorithm was not properly calculated, the API KEY is wrong or not valid

**Important:** because the signature string relies on a timestamp it is very important to have your server/system time synchronized with a NTP server. The web service allows only a 20 seconds difference between request and server time.

## API Reference

The following terminology will be used along the documentation:

- Sample Request - indicates a sample HTTP request sent to web service
- Sample Response - indicates a sample HTTP response returned by the web service
- Exception - the possible errors that the specific method can return
- API\_URL - the base URL for the API

Responses are shown here in XML format but the web service can also return data in the JSON format.

### Important:

- the header values in request/response samples do not contain valid data, this means that you cannot rely on X-PS-Auth header to contain a correct/valid signature
- the response body is sometimes truncated when it contains repetitive structures

## API Methods

This API has methods for retrieving details about advertisers and active campaigns, affiliate link generation and commission reporting.

### Advertisers

This method is used to retrieve details about active advertisers.

#### Advertisers listing

The details of active advertisers can be fetched by executing a GET request to the following URL:

- **[API\_URL]/affiliate-advertisers/**

#### Sample Request:

```
GET /affiliate-advertisers/ HTTP/1.1
Host: api.profitshare.bg
Date: Thu, 17 Jul 2012 14:52:54 GMT
X-PS-Client: test-account
X-PS-Accept: json
X-PS-Auth: 90a6d325e982f764f86a7e248edf6a660d4ee833
```

#### Sample Response:

```
HTTP/1.1 200 OK
```

```
Date: Thu, 17 Jul 2012 14:52:55 GMT
Server: profitshare.bg
Content-Length: 795
Connection: close
Content-Type: text/json
```

Response body:

```
{
  "result": [
    {
      "id": "100",
      "name": "Advertiser 1",
      "logo": "//profitshare.bg/files_shared/advertiser-logos/logo_1.png",
      "category": "Fashion",
      "url": "http://www.xxxx.bg"
    },
    {
      "id": "104",
      "name": "Advertiser 2",
      "logo": "//profitshare.bg/files_shared/advertiser-logos/logo_2.png",
      "category": "Fashion",
      "url": "http://www.yyyyyy.bg"
    },
    ...
  ]
}
```

## Campaigns

This method is used to retrieve details about active campaigns added by advertisers.

### Campaigns listing

This can be done by executing a GET request to the following address:

- **[API\_URL]/affiliate-campaigns/**

The results are paginated, each page having 20 results. In order to get page N, provide parameter page=N

### Sample Request:

```
GET /affiliate-campaigns/ HTTP/1.1
Host: api.profitshare.bg
Date: Tue, 17 Jul 2012 14:55:34 GMT X-PS-Client: test-account
X-PS-Accept: json
X-PS-Auth: 90a6d325e982f764f86a7e248edf6a660d4ee833
```

### Sample Response:

```
HTTP/1.1 200 OK
Date: Tue, 17 Jul 2012 14:55:35 GMT Server: Profitshare
Content-Length: 795
Connection: close
Content-Type: text/json
{
  "result": {
    "paginator": {
      "itemsPerPage": 20,
```

```

        "currentPage": 2,
        "totalPages": 2
    },
    "campaigns": [
    {
        "id": 118,
        "name": "Carti la eMAG",
        "commissionType": "CPS",
        "startDate": "2013-05-13 00:00:00",
        "endDate": "2013-12-31 00:00:00",
        "url": "http://www.emag.bg/carti/1/",
        "banners": {
            "468x60": {
                "width": 468,
                "height": 60,
                "src": "//profitshare.bg/images/advertiser_widgets/a.jpg"
            },
            "125x125": {
                "width": 125,
                "height": 125,
                "src": "//profitshare.bg/images/advertiser_widgets/s.swf"
            }
        }
    },
    ...
    ]
}
}

```

**Important:** Please be aware that our system also support flash files with .swf extension. In case you wish to integrate in html, please use the proper tags.

## Links

This method is used to generate affiliate links.

### Links add

This can be done by executing a POST request to the following address:

- **[API\_URL]/affiliate-links/**

The following parameters have to be sent in the POST request:

- links[N][name] (mandatory)
- link[N][url] (mandatory)

**Important:** Link URL and name should be url encoded.

```

POST /affiliate-links HTTP/1.1

Host: api.profitshare.bg
Date: Tue, 17 Jul 2012 14:55:34 GMT
X-PS-Client: test-account
X-PS-Accept: json
X-PS-Auth: 90a6d325e982f764f86a7e248edf6a660d4ee833

0%5Bname%5D=test+1&0%5Burl%5D=www.advertiser.bg&1%5Bname%5D=test+2&1%5Burl%5D=ww
ww.advertiser.bg

```

### Sample Response:

```
HTTP/1.1 201 No Content

Date: Tue, 17 Jul 2012 14:55:35 GMT
Server: Profitshare
Content-Length: 795
Connection: close
Content-Type: text/json

{
  "result": [
    {
      "name": "test 1",
      "url": "http://www.advertiser1.bg/landingpage1",
      "ps_url": "http://profitshare.bg/l/352101"
    },
    {
      "name": "test 2",
      "url": "http://www.advertiser2.bg/offers",
      "ps_url": "http://profitshare.bg/l/352102"
    },
    ...
  ]
}
```

**Important:** You can use an optional parameter for each affiliate link in order to get reporting based on this parameter. Please use the following URL structure: <http://profitshare.bg/l/{id}/{hash}>. For example:

- Link id = 123456
- Hash = qwexyz
- Affiliate link => <http://profitshare.bg/l/123456/qwexyz>

## Commissions

This method is used to retrieve details about conversions.

### Commission listing

This can be done by executing a GET request to the following address:

- **[API\_URL]/affiliate-commissions/**

The results can be filtered by providing additional parameters and ordered as well. The filters will be sent via the *filters* parameter and the order will be sent via the *order* parameter.

The supported filters will be sent as following:

- filters[<filter\_name>] = [filter\_value]

The *filter\_name* key will be validated and supports the following parameters:

- status - the status for the commission
- date\_from - for date\_added >= date\_from
- date\_to - for date\_added <= date\_to
- click\_hash – hash used as an optional parameter on affiliate links

The *order param* supports the following values:

- date
- commission
- The results are paginated, each page having 25 results. In order to get page N, provide parameter page=N

#### Sample Request:

```
GET /affiliate-commissions/? filters[status]=pending&filters[date_from]=2013-01-15&filters[date_to]=2013-01-16 HTTP/1.1
```

```
Host: api.profitshare.bg
Date: Tue, 17 Jul 2012 14:55:34 GMT X-PS-Client: test-account
X-PS-Accept: json
X-PS-Auth: 90a6d325e982f764f86a7e248edf6a660d4ee833
```

#### Sample Response:

```
HTTP/1.1 200 OK
Date: Tue, 17 Jul 2012 14:55:35 GMT Server: Profitshare
Content-Length: 795
Connection: close
Content-Type: text/json
```

```
{
  "result": {
    "current_page": 1,
    "total_pages": 2,
    "records_per_page": 25,
    "commissions": [
      {
        "order_id": 3000099327,
        "order_ref": "8657110",
        "order_status": "pending",
        "advertiser_id": 35,
        "hash": "",
        "order_date": "2013-11-01 14:20:37",
        "order_updated": "0000-00-00 00:00:00",
        "items_status": "pending|pending",
        "items_commission": "4.5965|56.3707",
        "items_commission_value": "3.00|3.00"
      },
      {
        "order_id": 3000099323,
        "order_ref": "45699",
        "order_status": "pending",
        "advertiser_id": 35,
        "hash": "",
        "order_date": "2013-11-01 13:53:00",
        "order_updated": "0000-00-00 00:00:00",
        "items_status": "pending",
        "items_commission": "5.4031",
        "items_commission_value": "1.00"
      },
      ...
    ]
  }
}
```

#### Result fields:



- "order\_id" => unique order identifier (profitshare);
- "order\_ref" => unique order identifier (advertiser);
- "order\_status" => general status of the order. May have one of the following values:
  - pending
  - approved
  - canceled
- „advertiser\_id" => unique advertiser id. Use method *Advertisers listing* to retrieve more details about advertisers;
- „hash" => optional parameter used in affiliate links. (For example: <http://profitshare.bg/l/{id}/{hash}>)
- „order\_date" => order date and time;
- „order\_updated" => last update date and time;

Each order has different items (products) that could be commissioned using different commission percents. Use the following fields to retrieve details about each item (separated by „|")

- „items\_status" => status of each order item separated by „|". May have one of the following values:
  - pending
  - approved
  - canceled
- „items\_commission" => value of each order item commission, in RON without VAT;
- „items\_commission\_value" => value of each order item commission percent;

## Products

This method is used to retrieve details about products.

### Products listing

This can be done by executing a GET request to the following address:

- **[API\_URL]/affiliate-products/**

The results can be filtered by providing additional parameters. The filters will be sent via the *filters* parameter.

The supported filters will be sent as following:

- filters[<filter\_name>] = [filter\_value]

The *filter\_name* key will be validated and supports the following parameters:

- advertisers - the status for the commission (filter value example: **45,56**)
- The results are paginated, each page having 20 results. In order to get page N, provide parameter page=N

### Sample Request:

```
GET /affiliate-products/? filters[advertisers]=45,41 HTTP/1.1
Host: api.profitshare.bg
Date: Tue, 17 Jul 2012 14:55:34 GMT X-PS-Client: test-account
X-PS-Accept: json
X-PS-Auth: 90a6d325e982f764f86a7e248edf6a660d4ee833
```

### Sample Response:

```
HTTP/1.1 200 OK
Date: Tue, 17 Jul 2012 14:55:35 GMT Server: Profitshare
Content-Length: 795
Connection: close
Content-Type: text/json
{
  "result":{
    "current_page":1,
    "total_pages":1,
    "records_per_page":20,
    "products":[
      {
        "link":"http://www.test.bg/test/test ",
        "name":"Laptop Lenovo",
        "image":"http://static.profitshare.bg/sfsd/sdfsdf.jpg",
        "price_vat":1299,
        "price":1047.58,
        "advertiser_id":41,
        "advertiser_name":"test.bg",
        "category_name":"Laptops"
      },
      {
        "link":"http://www.test.bg/test/test ",
        "name":"Laptop Lenovo",
        "image":"http://static.profitshare.bg/sfsd/sdfsdf.jpg",
        "price_vat":1299,
        "price":1047.58,
        "advertiser_id":41,
        "advertiser_name":"test.bg",
        "category_name":"Laptops"
      },
      ...
    ]
  }
}
```

### Result fields in products:

- "link" => link of product;
- "name" => name of product;
- "price vat" => price with vat of product
- "price" => price without vat of product
- "advertiser\_name" => advertiser name
- "category\_name" => category of product
- „advertiser\_id" => unique advertiser id. Use method *Advertisers listing* to retrieve more details about advertisers;

## Sample requests

### Getting the active advertisers list

```
<?php
$api_user = 'api_user';
$api_key = 'api_key';

$url = 'http://api.profitshare.bg/affiliate-advertisers/?';
$query_string = '';

$spider = curl_init();
curl_setopt($spider, CURLOPT_HEADER, false);
curl_setopt($spider, CURLOPT_URL, $url . $query_string);
curl_setopt($spider, CURLOPT_CONNECTTIMEOUT, 60);
curl_setopt($spider, CURLOPT_TIMEOUT, 30);
curl_setopt($spider, CURLOPT_RETURNTRANSFER, true);
curl_setopt($spider, CURLOPT_HTTPAUTH, CURLAUTH_ANY);
curl_setopt($spider, CURLOPT_USERAGENT, 'Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)');

$profitshare_login = array(
    'api_user' => $api_user,
    'api_key' => $api_key,
);

$date = gmdate('D, d M Y H:i:s T', time());
$signature_string = 'GETaffiliate-advertisers/?' . $query_string .
    '/' . $profitshare_login['api_user'] . $date;
$auth = hash_hmac('sha1', $signature_string, $profitshare_login['api_key']);

$extra_headers = array(
    "Date: {$date}",
    "X-PS-Client: {$profitshare_login['api_user']}",
    "X-PS-Accept: json",
    "X-PS-Auth: {$auth}"
);

curl_setopt($spider, CURLOPT_HTTPHEADER, $extra_headers);

$output = curl_exec($spider);
curl_close($spider);

echo $output;
```

### Advertiser products

```
<?php
$api_user = 'xxx';
$api_key = 'xxx';

$url = 'http://api.profitshare.bg/affiliate-products/?';
```

```

$query_string = 'page=1&filters[advertiser]=57323';

$spider = curl_init();
curl_setopt($spider, CURLOPT_HEADER, false);
curl_setopt($spider, CURLOPT_URL, $url . $query_string);
curl_setopt($spider, CURLOPT_CONNECTTIMEOUT, 60);
curl_setopt($spider, CURLOPT_TIMEOUT, 30);
curl_setopt($spider, CURLOPT_RETURNTRANSFER, true);
curl_setopt($spider, CURLOPT_HTTPAUTH, CURLAUTH_ANY);
curl_setopt($spider, CURLOPT_USERAGENT, 'Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)');

$profitshare_login = array(
    'api_user' => $api_user,
    'api_key' => $api_key,
);

$date = gmdate('D, d M Y H:i:s T', time());
$signature_string = 'GETaffiliate-products/?' . $query_string .
    '/' . $profitshare_login['api_user'] . $date;
$sauth = hash_hmac('sha1', $signature_string, $profitshare_login['api_key']);

$extra_headers = array(
    "Date: {$date}",
    "X-PS-Client: {$profitshare_login['api_user']}",
    "X-PS-Accept: json",
    "X-PS-Auth: {$sauth}"
);

curl_setopt($spider, CURLOPT_HTTPHEADER, $extra_headers);

$output = curl_exec($spider);
curl_close($spider);

echo $output;

```

## Link generator

```

<?php
$api_user = 'API_USER';
$api_key = 'API_KEY';

$url = 'http://api.profitshare.bg/affiliate-links/?';
$query_string = '';

$spider = curl_init();
curl_setopt($spider, CURLOPT_HEADER, false);
curl_setopt($spider, CURLOPT_URL, $url . $query_string);
curl_setopt($spider, CURLOPT_CONNECTTIMEOUT, 60);
curl_setopt($spider, CURLOPT_TIMEOUT, 30);
curl_setopt($spider, CURLOPT_RETURNTRANSFER, true);
curl_setopt($spider, CURLOPT_HTTPAUTH, CURLAUTH_ANY);
curl_setopt($spider, CURLOPT_USERAGENT, 'Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)');

$data = array();
$data[] = array(

```

```
'name' => 'API-emag',
'url' => 'http://www.emag.bg'
);

curl_setopt($spider, CURLOPT_POST, true);
curl_setopt($spider, CURLOPT_POSTFIELDS, http_build_query($data));

$profitshare_login = array(
    'api_user' => $api_user,
    'api_key' => $api_key,
);

$date = gmdate('D, d M Y H:i:s T', time());
$signature_string = 'POSTaffiliate-links/?' . $query_string .
'/' . $profitshare_login['api_user'] . $date;
$auth = hash_hmac('sha1', $signature_string, $profitshare_login['api_key']);

$extra_headers = array(
    "Date: {$date}",
    "X-PS-Client: {$profitshare_login['api_user']}",
    "X-PS-Accept: json",
    "X-PS-Auth: {$auth}"
);

curl_setopt($spider, CURLOPT_HTTPHEADER, $extra_headers);

$output = curl_exec($spider);
curl_close($spider);

echo '<pre>';
print_r($output);
```