

VideoBlocks API Documentation

Updated May 16, 2017

End Points

Videoblocks has three separate domains offering different types of content. The domains are:

- **VideoBlocks:** <https://api.videoblocks.com>
- **GraphicStock:** <https://api.graphicstock.com>
- **AudioBlocks:** <https://api.audioblocks.com>

They all share a common API, accessed using secure HTTPS GET requests. To avoid unnecessary redirects, include api with the domain.

Resources

Two resources are available to our partners:

- **Search:** *(GET)* [/api/v1/stock-items/search](https://api/v1/stock-items/search)
- **Download:** *(GET)* [/api/v1/stock-items/download/<stock item ID>/<downloader ID>](https://api/v1/stock-items/download/<stock item ID>/<downloader ID>)

Search

The search endpoint can be used to retrieve a list of stock items based on search parameters. The parameters are provided as URL parameters and can be any of the following:

Request Parameters

VideoBlocks

Parameter Name	Type	Description
keywords	string	Required. The search terms. Multiple terms can be separated with URL encoded spaces
page	int	Which page of results is being requested. (default: 1)
num_results	int	How many results per page to return. (default: 20)
contributor_id	string	The contributor portal ID. Searches for all items uploaded by contributor. Only used by VideoBlocks for now. If set, keywords is not required. Please reach out to contributor@videoblocks.com to get your contributor id.
sort	enum	Sort results by internal metrics. (default: most_relevant) Values: most_relevant, most_downloaded, highest_rated, most_recent, trending_now, undiscovered
content_type	enum	Filter results by content type. Multiple content types requested by providing a comma delimited list. (default: all) Values: all, footage, motionbackgrounds, templates
quality	enum	Filter results by quality (default: HD) Values: HD, SD, 4K

GraphicStock

Parameter Name	Type	Description
keywords	string	Required. The search terms. Multiple terms can be separated with URL encoded spaces
page	int	Which page of results is being requested. (default: 1)
num_results	int	How many results per page to return. (default: 20)
sort	enum	Sort results by internal metrics. (default: most_relevant) Values: most_relevant, most_downloaded, highest_rated, most_recent, trending_now, undiscovered)
content_type	enum	Filter results by content type. Multiple content types requested by providing a comma delimited list. (default: all) Values: all, photos, illustrations, vectors
categories	enum	Category filter. Multiple values may be provided in a comma delimited list. Values: design-elements, objects, business, people, holidays-and-celebrations, nature-and-wildlife, buildings-and-landmarks, sports-and-outdoors, fictional, travel-and-transportation, health-and-science, backgrounds, cultural, beauty-and-fashion
orientation	enum	Orientation filter. Values: all, portrait, square, landscape
has_transparency	bool	Transparency filter. If true, then content without an alpha channel will be excluded
color	string	A 6 character hexadecimal value of a color (e.g. #BADA55). When provided, only the content that has the provided color will be included

AudioBlocks

Parameter Name	Type	Description
keywords	string	Required. The search terms. Multiple terms can be separated with URL encoded spaces
page	int	Which page of results is being requested. (default: 1)
num_results	int	How many results per page to return. (default: 20)
sort	enum	Sort results by internal metrics. (default: most_relevant) Values: most_relevant, most_downloaded, highest_rated, most_recent, trending_now, undiscovered)
content_type	enum	Filter results by content type. Multiple content types requested by providing a comma delimited list. (default: all) Values: all, music, loops, sfx
min_duration	int	Minimum duration filter. When this parameter is provided any content having a duration less than the provided value will be excluded. (default: 1)
max_duration	int	Maximum duration filter. When this parameter is provided any content having a duration greater than the provided value will be excluded. (default: 1000)
genres	enum	Genre filter. Applies only when the content_type is set to music. Multiple values may be provided in a comma delimited list. Values: alternative-rock, ambience, blues, bumpers-and-stingers, cinematic, classic-rock, classical, corporate, country, dramatic, easy-listening, electronic, funk, guitar, hard-rock-and-heavy-metal, hip-hop, holiday, horror, instrumental, island, jazz, percussion, playful-and-comedic, pop, punk-rock, religious, romantic, techno, uplifting-and-epic, world
moods	enum	Mood filter. Applies only when the content_type is set to music. Multiple values may be provided in a comma delimited list. Values: aggressive, epic-inspiring, happy-upbeat, playful-silly, relaxing, sad, sentimental,

		suspenseful
instruments	enum	Instrument filter. Applies only when the content_type is set to music. Multiple values may be provided in a comma delimited list. Values: orchestra, guitar-instrument, brass, percussion-instrument, vocals, woodwind, piano

Response

The response will include an array of search results, with content that varies depending on the site which is being searched.

- **success** - flag that will be set to true
- **totalSearchResults** - total results for the given search, useful for pagination
- **info** - array of search result information that includes:
 - **id** - the unique identifier of the result
 - **title** - short title of the result
 - **details_url** - link to the details page for that item
 - **preview_url** - link to a preview file
 - **thumbnail_url** - link to a thumbnail image
 - **download_formats** - available download formats for the asset

Example VideoBlocks search request (excludes authentication related URL parameters):

```
https://api.videoblocks.com/api/v1/stock-items/search?keywords=plane&page=1&num_results=2
```

Example VideoBlocks search response:

```
{
  "success":true,
  "totalSearchResults":100,
  "message":null,
  "info":[
    {
      "id":9328,
      "title":"Plane Taking Off Extremely Close Up",
      "details_url":"https://videoblocks.com/video/plane-close/",
      "preview_url":"https://abc.cloudfront.net/video/preview/plane.mp4",
      "thumbnail_url":"https://abc.cloudfront.net/video/thumbnail/plane.jpg",
```

```

        "duration":10,
        "download_formats":[MP4, MOV],
    },
    {
        "id":9276,
        "title":"Plane Taking Straight Off",
        "details_url":"https://videoblocks.com/video/plane-str/",
        "preview_url":"https://abc.cloudfront.net/video/preview/plane-str.mp4",
        "thumbnail_url":"https://abc.cloudfront.net/video/thumbnail/str.jpg",
        "duration":13,
        "download_formats":[MP4]
    }
]
}

```

Example GraphicStock search request (excludes authentication related URL parameters):

https://api.graphicstock.com/api/v1/stock-items/search?keywords=plane&page=1&num_results=2

Example GraphicStock search response:

```

{
  "success":true,
  "totalSearchResults":100,
  "info":[
    {
      "id":43956,
      "description":"Commercial Jet Plane Airliner On Isolated Background",
      "details_url":"https://graphicstock.com/stock-image/jet-plane",
      "type":"Illustration",
      "title":"Jet Plane",
      "thumbnail_url":"https://abc.cloudfront.net/thumbnail/jet-plane.jpg",
      "preview_url":"https://abc.cloudfront.net/preview/jet-plane.jpg",
      "download_formats":[EPS, JPG]
    },
    {
      "id":2546,
      "description":"Silhouette Plane In The Air",
      "details_url":"https://graphicstock.com/stock-image/silhouette-plane",
      "type":"Illustration",
      "title":"Silhouette Plane",
      "thumbnail_url":"https://abc.cloudfront.net/thumbnail/silhouette-plane.jpg",
      "preview_url":"https://abc.cloudfront.net/preview/silhouette-plane.jpg",
      "download_formats":[EPS]
    }
  ]
}

```

```
]
}
```

Example AudioBlocks search request (excludes authentication related URL parameters):

```
https://api.audioblocks.com/api/v1/stock-items/search?keywords=plane&page=1&num_results=2
```

Example AudioBlocks search response:

```
{
  "success":true,
  "totalSearchResults":100,
  "info":[
    {
      "id":50060,
      "details_url":"https://audioblocks.com/stock-audio/plane-take-off.html",
      "type":"sfx",
      "duration":23,
      "bpm":87,
      "title":"Awac Plane Take Off",
      "thumbnail_url":"https://abc.cloudfront.net/sfx/thumbs/plane-take-off.png",
      "preview_url":"https://abc.cloudfront.net/sfx/mp3preview/plane-take-off.mp3",
      "download_formats":[MP3]
    },
    {
      "id":51810,
      "details_url":"https://audioblocks.com/stock-audio/plane-crash.html",
      "type":"sfx",
      "duration":13,
      "bpm":148,
      "title":"Single Engine Plane Crash",
      "thumbnail_url":"https://abc.cloudfront.net/sfx/thumbs/plane-crash.png",
      "preview_url":"https://abc.cloudfront.net/sfx/mp3preview/plane-crash.mp3",
      "download_formats":[WAV]
    }
  ]
}
```

Download

The download endpoint is used to retrieve a signed download url that allows downloading directly from our content distribution network. The url expires after 20 minutes. This request takes no URL parameters, but instead has the parameters as part of the requested resource:

- **stock item ID** - id of the the item to be downloaded. This is provided in the search results API response.
- **downloader ID** - unique identifier for the downloader. This is to record which user on the partner site is doing the download. Helpful for debugging and analyzing usage. Typically this would be a unique, non personally identifiable id corresponding to the user on the partner site doing the downloading. For example the partner's database id for the user.

Currently only one download format is provided. Videos will provide the mp4 format, audio will provide the wav format, and graphics the jpg format. The response will include:

- **success** - flag that will be set to true
- **info** - contains information about the download, including:
 - **url** - signed url to download the item

Example download request (Excludes authentication related URL parameters):

<https://api.videoblocks.com/api/v1/stock-items/download/9328/1234>

Response

Example download response:

```
{
  "success":true,
  "info":
    {
      "url":"https://abc.cloudfront.net/video/content/hd-las-vegas-bellagio_wk
urxkoybmov__D.mp4?response-content-disposition=attachment&Policy=eyJTdGF
0ZW11bnQiO1t7I1Jlc291cmNlIjoiaHR0cHM6Ly9kMnY5eTBkdWtyNm1xMi5jbG91ZGZyb25
0Lm5ldC92aWRlby9jb250ZW50L2hkLWxhcy12ZWdhcy1iZWxsYWdpb193a3VyeGtveWJtb3Z
fX0QubXA0P3Jlc3BvbnNlLWVbnRlbnQtZGlzcG9zaXRpb249YXR0eWwNobWVudCIsIknvbmR
pdGlvbI6eyJEYXRlTGZvc1RoYW4iOnsiQVdtOkVwb2NoVGltZSI6MTQ1ODUxOTU4N319fV1
```



```

    9&Signature=RS6mw6DFxxITISMwgzLqzSaWhD8tdYMkSc7H3nphChq~UPmPRpTb1LFGeoUc
    j9g4gFLAw6eUiI1G0YdoH~19EP5hLS6VuPdsbJS2uiDQ1QR0sNuEbd0~GoBxUZbvXsU-7QDG
    fwduB4BfBt-sz89dRAaBNlWgbd~mowutqWvK176y9NLN6j90vNCR2GI6QcDHBA92HRp2yjuh
    9CfqRJC2sDIYIh5Nkc5A3uiBvA5kaaH7Aegkavbf4RPvw6gPaskjb6JNEBxtORvqvYX1Rkrj
    YJWGRXEg8cBmExlk4o2pbLH25yFdoJk5Q4WhEXQpfa6SsroHCwtVCeMF556KRc-Z2A__&Key
    -Pair-Id=APKAIAQ2JPZII02PHR7Q"
  }
}

```

Limits

API calls are limited at the resource level, and can vary for each partner as they are formulated in the agreement with VideoBlocks. For example search API calls can have a different limit than download API calls.

Errors

If an error is encountered, the status code will be 40X and the response will include:

- **success** - flag that will be set to false
- **message** - short message describing the error
- **description** - longer description of the error
- **code** - numerical reference code should you need support from the VideoBlocks team

Example error response:

```

{
  "success":false,
  "message":"Exceeded allowed API requests",
  "description":"You have hit or exceeded the number of daily API requests allowed.",
  "code":1004
}

```


Authentication

VideoBlocks uses a keyed-hash message authentication code (HMAC) to validate API requests. Each partner is assigned a public key and a secret key. Each API request requires three additional url parameters:

- **APIKEY** - the partner's public key
- **EXPIRES** - an expiration time in seconds since the Unix epoch (January 1, 1970)
- **HMAC** - authentication code

Currently the EXPIRES parameter is unsupported, and a 120 second expiration window is assigned to all requests. Until further notice, use the current time for the value for EXPIRES.

The authentication code is generated by using the SHA-256 hmac algorithm with the requested resource as the data and the secret key concatenated with the expiration time as the key. The hash is expected in hexadecimal format.

Sample implementation of an HMAC signed request in PHP:

```
// authentication keys provided by VideoBlocks
$publicKey = '01234567890abcdef';
$privateKey = 'fedcba09876543210';

// Generate HMAC for VideoBlocks API Authentication
$time = time();
$resource = '/api/v1/stock-items/search/';
$keyword = 'plane';
$hmac = hash_hmac('sha256', $resource, $privateKey . $time);
$url = "https://api.videoblocks.com" . $resource . "?APIKEY=" . $publicKey . "&HMAC="
. $hmac . "&EXPIRES=" . $time . "&keywords=" . $keyword;

// make the API request
$session = curl_init();
curl_setopt($session, CURLOPT_URL, $url);
curl_setopt($session, CURLOPT_SSL_VERIFYPEER, false);
$result = curl_exec($session);
curl_close($session);
print_r($result);
```

Sample implementation of an HMAC signed request in Node.js:

```
// required modules
var crypto = require('crypto');
var url = require('url');
var request = require('request'); // install using npm

// authentication keys provided by VideoBlocks
var VB_API_PUBLIC_KEY = '01234567890abcdef';
var VB_API_PRIVATE_KEY = 'fedcba09876543210';

var resource = '/api/v1/stock-items/search/';
var videoBlocksUrl = 'https://api.videoblocks.com'+resource;

// Generate HMAC for VideoBlocks API Authentication
var unixTimeInSeconds = Math.floor(Date.now() / 1000);
var hmac = crypto.createHmac('sha256', VB_API_PRIVATE_KEY + unixTimeInSeconds);
hmac.update(resource);
var hmacString = hmac.digest('hex');

// set up query options
var urlParms = {
  //Search params
  keywords: 'plane',
  //HMAC params
  EXPIRES: unixTimeInSeconds,
  APIKEY: VB_API_PUBLIC_KEY,
  HMAC: hmacString
};

// generate the request module options
var requestOptions = {
  url: videoBlocksUrl,
  qs: urlParms,
  useQuerystring: true,
  json: true,
  method: 'GET'
};

// make the API request
request(requestOptions, function(err, httpResponse, response) {
  if (!err && httpResponse.statusCode === 200) {
    console.info('Received VideoBlocks API response:', response);
  } else {
    console.error('Error communicating with VideoBlocks API');
  }
});
```