blekko

Slash the web!

blekko.com

API Spec Reference

November 30, 2011

Revision 1.0.2

Table of Contents

Table of Contents 2
Introduction..... 3
Asking for an API Auth Key 3
Use Cases..... 3
    Meta-Search or Custom search engines 3
    Mobile Apps..... 4
    Browser Toolbars 4
    SEO (search engine optimization) Tools..... 4
    Embedding blekko search on your website 4
Search Query API 5
Pagestats API (Toolbars)..... 6
Slashtag API (plus login/logout)..... 7
    Logging in via the API 7
    Finding the current username..... 7
    Adding to an existing slashtag via the API..... 8
    Creating a new slashtag 8
    Removing URLs from a slashtag 9
    Listing sites in a slashtag 9
    Deleting a slashtag 10
Advice to Implementers 11
Terms of Service 12
Revision History 12

Introduction

The blekko search engine is all about transparency. Along those lines, we offer a full-featured API (application programming interface) to access blekko's search results and other data, and also to manipulate slashtags.

Please note that you can do any query through https:// if your libraries support it; your users will thank you for improving their privacy. Logging in to manipulate slashtags requires https.

Asking for an API Auth Key

We are not currently restricting API access to only people with API keys. However, you should still ask for one. If we have your email address, and we become unhappy with the volume or some other aspect of your queries, we will be able to contact you. If we can't contact you, our only recourse is to cut you off. Please email us at **apiauth@blekko.com**, and ask for an API auth key. Then include it as a cgi argument in all your blekko queries. We'll send future API update information to your email address.

We ask that you limit searches and API calls to blekko to 1 per second. Contact us first if you'd like to send us more traffic than that.

Using https: all the time will help keep your api auth key secret.

Use Cases

There are a few main use cases for accessing the blekko API.

Meta-Search or Custom search engines

A meta-search engine, such as the one at ixquick.com, sends queries to a variety of search engines, and displays re-ordered and combined results. Our Query API provides the appropriate data. Note that we have features such as suggested slashtags and autoslashtags, which are worth showing to the user.

Mobile Apps

A mobile app might want to use blekko to provide searches of a subset of the web. For example, a web app focusing on a local region might want to provide full-text search of local websites. To do this, first create a slashtag using blekko's website or the slashtag API, and then use the query API to fetch results, adding your slashtag to the query terms.

Browser Toolbars

A browser toolbar, such as the blekko Firefox toolbar, might wish to display information about the current website, such as the number of inlinks and the site rank. This information is provided by the pagestats API. The blekko toolbar also allows the user to tag the current page, or mark it as spam. This functionality is provided by the slashtag API.

SEO (search engine optimization) Tools

blekko has a lot of data on our /seo and other webpages of use for SEO. Unfortunately, at this moment most of this data is not available in JSON form. Please stay tuned. We also have a separate API for collecting inbound links, described in a separate document.

Embedding blekko search on your website

If you make a slashtag that users of your website would want to use, the easiest way to present this to your users is by having a search box on your website which initiates a search (with your slashtag added) at blekko. The HTML+javascript to do this can be gotten by viewing any slashtag ("/view /foo") and clicking on the funny icon with "< >" in it.

Search Query API

The safest way to call the query API is using a (properly escaped) CGI argument,

http://blekko.com/ws/?q=<QUERY>&auth=<APIKEY>

Argument Name	Description
q	the query terms, with CGI escaping
auth	your API Auth key
p	page of results to fetch (optional)

For example, to run the query [cure for headaches], use

http://blekko.com/?q=cure+for+headaches&auth=<APIKEY>

This query will return HTML by default. You can ask for json output using the /json slashtag, and xml output using the /rss slashtag.

JSON example:
http://blekko.com/?q=cure+for+headaches+/json&auth=<APIKEY>

If you'd like more than the default 20 results, add /ps=100 for 100 results per page (100 is the max.) This is more efficient for us than requesting 100 results 20 at a time. To get the Nth page of results, add &p=<N>. So, to get the second hundred results:

http://blekko.com/?q=cure+for+headaches+/json+/ps=100&auth=<APIKEY>&p=1

The JSON reply to queries contains some useful information which you may want to highlight or use. If we autofired a slashtag, such as /health for the query [cure for headaches], auto_slashtag in the reply will be set to the slashtag name, and the final query (adding the slashtag, and perhaps dropping some words) will be in auto_slashtag_query. Next, if we have any suggested slashtags, they will be found in the list sug_slash.

An interesting slashtag you may wish to use is /fastdate. If you want to find "real time" results for a search, /fastdate quickly computes an approximate answer using little of blekko's resources.

If you are using /date, note that in some cases /date /more gives a much better result than plain /date.

SEO Information API

We make a large amount of information public via our /seo slashtag, and the slashtags further used on the /seo page. Alas, at this moment the /json slashtag is ignored for these queries, but we hope to fix that soon.

If you are scraping our html, please use your APIKEY and the /ps=100 slashtag to reduce the number of queries.

Due to the large number of scrapers sending us queries via botnets, most of our SEO data slashtags now require a login or an APIKEY.

We have a separate API for fetching lots of inbound links in a single call. This API is described in a separate document. This API is limited by IP address – you will need to send us the IP addresses of your servers to be authorized by us. If your app is distributed, you will need to run all your blekko inbound API queries through one of your servers.

Pagestats API (Toolbars)

The pagestats API returns appropriate data for toolbars:

https://blekko.com/api/pagestats?url=http://example.com&auth=<APIAUTH>

Argument Name	Description .
url	full URL of some webpage, including http:// or https://

The returned data is always json, and looks like:

```
{"adsense":null,"cached":false,"dup":false,"host_inlinks":672,"host_rank":430.000
672,"ip":null,"rss":null}
```

The elements adsense, cached, dup, ip, and rss correspond to the buttons on blekko's search results. You can only ask for info on one URL at a time through this interface.

Slashtag API (plus login/logout)

You can create and add to slashtags via the web as well as via this API – note that you can upload a text or OPML file with large numbers of URLs via the website interface.

With either method, there’s a limit of adding 5,000 URLs in one transaction, a limit of 25,000 URLs in one slashtag, and a limit of 100,000 URLs in a single search, if you either use nested slashtags or multiple slashtags.

This API is somewhat unusual in that the calls below are part of our AJAX interface, and as such the return values are not always JSON.

Logging in via the API

If you want to create/add to slashtags via the API, you will need to log in to an existing blekko account. You can steal your own A cookie from your browser cache (eventually we'll fix this replay vulnerability) by, for example in Firefox, logging in and then looking at Tools -> Page Info -> Security -> View Cookies.

Better, you can log in via

https://blekko.com/login?u=<USERNAME>&p=<PASSWORD>&auth=<APIAUTH>

Argument Name	Description
u	existing blekko username
p	password

This API call must use https. The return value is always JSON. On success, blekko will return a JSON response including 'status':1, and whose header contains a Set-Cookie with the login cookie in it. The login cookie's name is "A". If USERNAME or PASSWORD is wrong, the response body will not contain ‘status’:1.

Finding the current username

In situations where the user has already logged into blekko in their browser, for example in a browser toolbar, it is useful to find out the current username. This can be done by:

https://blekko.com/api/userinfo

This call returns JSON, with one element, username.

Adding to an existing slashtag via the API

Once you have your A cookie, you can add to existing slashtags:

http://blekko.com/tag/add?name=<SLASHTAG>&submit=update&urls=<LIST>&auth=<APIAUTH>

Argument Name	Description .
submit=update	
name	the full name of a slashtag, e.g. /greg/spam
urls	A list of URLs, separated by %0A

URLs added to a slashtag can be the names of servers, full URLs, a partial URL path (such as espn.com/nfl/), or a URL pattern (such as stackoverflow.com/*perl)

The usual limits of 5,000, 25,000, and 100,000 (as explained above) apply to this API call.

An empty return value indicates success. A text error is returned for failures, or, a redirect if the user is not logged in.

Creating a new slashtag

Creating a new slashtag is similar to updating an old tag, simply substitute submit=create for submit=update:

http://blekko.com/tag/add?name=<SLASHTAG>&submit=create&urls=<LIST>&auth=<APIAUTH>

Argument Name	Description .
submit=create	
name	the full name of a slashtag, e.g. /greg/spam
urls	a list of URLs separated by %0A

desc	(optional) short string describing the slashtag

If the tag already exists, the response currently looks something like "slashtag /USERNAME/TAGNAME already exists", and the slashtag remains unchanged.

When creating a tag, &urls= is optional; without urls, you will get an empty slashtag.

You can optionally add &desc=<DESCRIPTION> to add a description to your new slashg.

The return values for this call are the same as for adding to an existing slashtag.

Removing URLs from a slashtag

To remove URLs from an existing slashtag, use

http://blekko.com/tag/edit?submit=1&type=del&name=<SLASHTAG>&urls=<LIST>&auth=<APIAUTH>

Argument Name	Description .
submit=1	
type=del	
name	full name of a slashtag, e.g. /greg/spam
urls	list of URLs to delete, separated by %0A

The return value for this call is always a blob of JavaScript. If there was an error, the string “alertMsgErr” will be present in the JavaScript.

Listing sites in a slashtag

You can list the sites belonging to an existing slashtag with

http://blekko.com/tag/view?name=<SLASHTAG>&format=json&auth=<APIAUTH>

Argument Name	Description .
name	full name of a slashtag, e.g. /greg/spam
format	=json or =text

format=text gives a simple list of URLs. format=json gives both the full URL and a version formatted for display.

The return value (for either format of this call) is text or json on success, and HTML on error. We recommend considering either an initial "<" or the 6-character string "Error:" to indicate an error.

Deleting a slashtag

A slashtag can be deleted using:

http://blekko.com/tag/delete?submit=1&name=<SLASHTAG>&auth=<APIAUTH>

Argument Name	Description .
submit=1	
name	full name of slashtag, e.g. /greg/spam

The return value for this operation is similar to removing websites from a slashtag; the string "alertMsgErr" will be present if there is an error.

[Note: at the moment, this call always fails. We will issue updated documentation when it is fixed.]

Advice to Implementers

There is a API client reference implementation for this interface in Perl at cpan.org – it is named Webservice::Blekko. It contains tests for the above-mentioned somewhat idiosyncratic return values. We recommend that implementations in other languages use similar names for the various functions:

query
pagestats
login
logout
user_info
create_slashtag
add_urls
list_urls
delete_urls
remove_slashtag

We also recommend that API clients enforce the 1 QPS (query per second) limitation discussed above. The perl code does this by remembering the time of the last query, and sleeping an appropriate amount of time if a subsequent call occurs too quickly. It applies this limit to every API call, not just search queries.

Terms of Service

blekko's terms of service may be reviewed at:

https://blekko.com/ws/+/terms

and…

https://blekko.com/ws/+/apiterms

.

We are currently only issuing formal written authorization to do things prohibited by the above terms to very large users of the API. If you are a small user of the API, are well behaved (e.g. not flooding us with queries), and we know your email address (from when you applied for an API auth key, see above), we will to attempt to contact you and discuss your usage patterns if needed.

Revision History

1.0 March 1, 2011
1.0.1 April 10, 2011 – added /ws/ to query url
1.0.2 November 30, 2011 – added note about inbound link API, and the need for login or auth to use SEO slashtags.