SpiderFoot HX API 1.3

This document describes version 1.3 of the SpiderFoot HX API, designed to provide developers the ability to directly interact with the SpiderFoot back-end without going through a web browser. This gives you the ability to automate running scans, retrieving their results, checking their status and more.

The API is using simple HTTP methods returning data in JSON format, therefore you can use it from a language and environment of your choice.

Usage Guidelines

- All requests are to be made over HTTPS - both GET and POST are supported.
- All responses are in JSON format.
- You must supply an API key with all queries. This key can be found in your account settings and can be re-generated as needed.
- Re-generating the API key immediately voids the previous key.
- Dates returned are in UTC, formatted as YYYY-MM-DD HH:MM:SS.

Calling the API

To test the API, you can use cURL:

```
~$ curl --blah \
'https://example.hx.spiderfoot.net/api?apikey=YOUR_API_KEY&func=FUNC_NAME&arg1=X&arg2=Y'
```

Where *YOUR_API_KEY* is the API key obtained from your SpiderFoot HX interface, *FUNC_NAME* above is the name of the API function being called (see the API Calls section below) and *arg1*, *arg2*, etc. are the arguments expected by that function.

Refer to the examples in each of the API calls documented below to get a better idea of how this works in practice.

Errors

Any failed API call will result in a JSON formatted tuple consisting of two fields - the status code (always ERROR) and the second field containing further information about the error, looking similar to the following:

```
['ERROR', 'Error message here']
```

For instance, if you were to supply an invalid API key for your request, it would look something like this:

```
~$ curl 'https://example.hx.spiderfoot.net/api?func=scanlist&apikey=somethingmadeup'
["ERROR", "Invalid API Key."]
```

API Calls

SpiderFoot Information

Name	eventtypes
Description	Returns a list of all event types in SpiderFoot.
Arguments	*None*
Returns	Array of arrays representing the name and description of every event type.

Example	`~$ curl \`
	`'https://example.hx.spiderfoot.net/api?func=eventtypes&apikey=YOUR_API_KEY'`
	`[`
	`  [`
	`  "ACCOUNT_EXTERNAL_OWNED",`
	`  "Account on External Site"`
	`  ],`
	`...`
	`]`

Name	modules
Description	Returns a list of modules available in SpiderFoot.
Arguments	*None*
Returns	Array of hashes for each module, representing the short name and description of each module.

Example	`~$ curl \`
	`'https://example.hx.spiderfoot.net/api?func=modules&apikey=YOUR_API_KEY'`
	`[`
	`  {`
	`  "name" : "sfp_abusech",`
	`  "descr" : "Check if a host/domain, IP or netblock is malicious according to abuse.ch."`
	`  },`
	`...`
	`]`

Scan Management

Name	scanstart
Description	Initiates a SpiderFoot scan against a provided target or set of targets.
Arguments	*name* - The name you wish to give the scan.
	target - The actual target(s), being an IP address, hostname, domain name, subdomain name or netblock/subnet. For multiple targets, simply comma-separate them.
	modules - A comma-separated list of modules that you want to use for this scan. The names of all modules and their functionality is available here: http://www.spiderfoot.net/documentation/#module-list
	callback - The URL to call after the scan has completed. A JSON object is POSTed to this URL in the format of { id: <scan instance ID>, status: <status> }
Returns	Tuple of the status and the ID of the scan started.

Example	`~$ curl \`
	`'https://example.hx.spiderfoot.net/api?func=scanstart&apikey=YOUR_API_KEY_HERE&name=MyScan&target=example.com&modules=sfp_spider,sfp_dns'`
	`["SUCCESS",`
	`"bd499640df058efe3bfeb5e599c34cd0bc00cc49a37b66c4fa69b698b15bb781"]`

Name	scanstop
Description	Requests that a running scan be aborted.
Arguments	*id* - The ID of the scan (*not* the name).
Returns	Tuple representing the success of the call made. **Note:** Even if the call returns a *SUCCESS* (meaning the scan was stopped), a scan may not actually be stopped but in the *ABORT-REQUESTED* state as SpiderFoot attempts to cleanly abort the scan. If you want to be certain that the scan is stopped, you should follow your *stopscan* call with a *scanstatus* call (see below).

Example	`~$ curl \`
	`'https://example.hx.spiderfoot.net/api?func=scanstop&apikey=YOUR_API_KEY&id=bd499640df058efe3bfeb5e599c34cd0bc00cc49a37b66c4fa69b698b15bb781'`
	`["SUCCESS", "Scan abort successfully started."]`

Scan Information

Name	scanlist
Description	Lists all scans that are in your SpiderFoot instance.
Arguments	*None required.*
Returns	An array of hashes for each scan with the following fields: - *scan_id*: The ID of the scan. - *scan_target*: The target(s) of the scan, i.e. a comma-separated list of domain names, IP addresses, etc. that were targeted. - *scan_name*: The name that was given to the scan when created. - *start_time*: The time the scan started running against the target. - *created_time*: The time the scan was created in the SpiderFoot database. - *end_time*: The time the scan stopped running (either from having completed, failed or being aborted.) - *result_count*: The number of elements generated from this scan. - *scan_status*: The status of the scan, which may be *FINISHED, ABORTED, ABORT-REQUESTED, ERROR-FAILED, STARTED, STARTING, RUNNING*. - *scan_method:* The method used to initiate the scan, which may be UI, API or SCHEDULED. - *schedule_id:* The ID number of the schedule that initiated this scan, or null if the scan was initiated via API or UI.

Example	`~$ curl \`
	`'https://example.hx.spiderfoot.net/api?func=scanlist&apikey=YOUR_API_KEY'`
	`[{"scan_id": "bd499640df058efe3bfeb5e599c34cd0bc00cc49a37b66c4fa69b698b15bb781", "scan_target": "binarypool.com", "scan_name": "MyScan", "end_time": "2015-09-27 08:17:07", "created_time": "2015-09-27 08:16:11", "result_count": 320, "scan_method": "UI", "start_time": "2015-09-27 08:16:11", "scan_status": "FINISHED"}]`

Name	scanstatus
Description	Returns status information about a given scan.
Arguments	*id* - The ID of the scan.
Returns	A hash with the following fields: - *scan_target*: The target(s) of the scan, i.e. a comma-separated list of domain names, IP addresses, etc. that were targeted. - *scan_name*: The name that was given to the scan when created. - *start_time*: The time the scan started running against the target.

	- *created_time*: The time the scan was created in the SpiderFoot database. - *end_time*: The time the scan stopped running (either from having completed, failed or being aborted.) - *scan_status*: The status of the scan, which may be *FINISHED, ABORTED, ABORT-REQUESTED, ERROR-FAILED, STARTED, STARTING, RUNNING*.

Example	`~$ curl \`
	`'https://example.hx.spiderfoot.net/api?func=scanstatus&apikey=YOUR_API_KEY&id=bd499640df058efe3bfeb5e599c34cd0bc00cc49a37b66c4fa69b698b15bb781'`
	`{"scan_target": "binarypool.com", "scan_name": "MyScan", "end_time": "2015-09-27 08:17:07", "created_time": "2015-09-27 08:16:11", "start_time": "2015-09-27 08:16:11", "scan_status": "FINISHED"}`

Name	scansummary
Description	Returns a summary of the results for a scan.
Arguments	*id* - The ID of the scan.
Returns	A hash with the following fields: - *type_id*: The internal identifier for the element type, e.g. *IP_ADDRESS*. - *type_name*: The descriptive name for the element type, e.g. *IP Address*. - *last_updated*: When a data element was last found for this type. - *total_elements*: The total number of data elements of this type. - *unique_elements*: The number of unique data elements of this type.

Example	`~$ curl \`
	`'https://example.hx.spiderfoot.net/api?func=scansummary&apikey=YOUR_API_KEY&id=bd499640df058efe3bfeb5e599c34cd0bc00cc49a37b66c4fa69b698b15bb781'`
	`[{"total_elements": 6, "type_name": "Affiliate - Internet Name", "last_updated": "2015-09-27 08:16:22", "unique_elements": 6, "type_id": "AFFILIATE_INTERNET_NAME"}, {"total_elements": 4, "type_name": "Affiliate - IP Address", "last_updated":<... snipped ...>"unique_elements": 11, "type_id": "TARGET_WEB_CONTENT"}]`

Name	scanlog
Description	Returns all the log messages generated during a scan.
Arguments	*id* - The ID of the scan.
Returns	An array of hashes for each log message with the following fields: - *generated*: The date and time the log message was generated.

	- *component*: The SpiderFoot component or module that generated the log message. - *log_type*: The type of log (INFO, ERROR, DEBUG or WARNING). - *log_message*: The log message itself.
Example	~$ curl \ 'https://example.hx.spiderfoot.net/api?func=scanlog&apikey=YOUR_API_KEY&id=bd499640df058efe3bfeb5e599c34cd0bc00cc49a37b66c4fa69b698b15bb781' [{"log_message": "Scan [bd499640df058efe3bfeb5e599c34cd0bc00cc49a37b66c4fa69b698b15bb781] completed.", "component": "SpiderFoot", "log_type": "STATUS", "time": "2015-09-27 08:17:07"}, {"log_message": "Unable to IPv6 resolve binarypool.com ([Errno -2] Name or service not known)", "component": "modules.sfp_dns", "log_type": "INFO", "time": "2015-09-27 08:17:07"}, <... snipped ...>{"log_message": "Scan [bd499640df058efe3bfeb5e599c34cd0bc00cc49a37b66c4fa69b698b15bb781] initiated.", "component": "SpiderFoot", "log_type": "STATUS", "time": "2015-09-27 08:16:11"}]

Scan Result Data

Name	scanresults
Description	Returns all results from a scan limited by either element type or module. If over 10,000 records exist in the result-set, you must use paging.
Arguments	*id* - The ID of the scan.
	type - The element type ID, being one of those listed here: http://www.spiderfoot.net/documentation/#data-elements
	module - The name of the module that generated the data element (e.g. sfp_spider)
	paged - 1 (true) or 0 (false) if you want the results broken up into pages. If set to 1, then you must also specify a *start* to represent the record offset and *count* for the number of records to return from that offset.
	start - The starting record number to return results for.
	count - The number of records to return.
Note	At least a type or module must be specified, or both.
Returns	An array of hashes for each data element with the following fields: - *id*: Internal ID of this data element. - *source*: The source data from which this data element was found. - *date_found*: Date the data element was found. - *data*: The actual data itself, e.g. the IP address, hostname, etc. - *module*: Module that generated this element - *confidence*: Confidence-score (not yet used) - *visibility*: Visibility-score (not yet used) - *risk*: Risk-score (not yet used) - *fp*: Flag (0 or 1) indicating whether this data element has been flagged as a false positive by the user. - *parent_fp*: Flag (0 or 1) indicating whether the data element that generated this data element was set as a false positive by the user. - *notes*: User-supplied notes about this data element. - *parent_notes*: User-supplied notes about the data element that generated this data element.
Example 1 (non-paged - returns all results)	~$ curl \ 'https://example.hx.spiderfoot.net/api?func=scanresults&apikey=YOUR_API_KEY&id=bd499640df058efe3bfeb5e599c34cd0bc00cc49a37b66c4fa69b698b15bb781&type=IP_ADDRESS' [{"fp": 0, "confidence": 100, "risk": 0, "parent_fp": 0, "notes": null, "parent_notes": null, "visibility": 100, "source": "binarypool.com", "date_found": "2015-09-27 08:17:08", "module": "sfp_dns", "data":

	"104.236.67.238", "id": "c202875808c4a99f8d04851df4507520f0ce15f7aee2880b5ce907f6da7fcbac"}, {"fp": 0, "confidence": 100, "risk": 0, "parent_fp": 0, "notes": null, "parent_notes": null, "visibility": 100, "source": "www.binarypool.com", "date_found": "2015-09-27 08:16:16", "module": "sfp_dns", "data": "104.236.67.238", "id": "1c9a0932b44055c4b4dfd56833b100f16621f96ed25d7483a3f012a68742f3ed"}]
Example 2 (paged - returns the first 100 results)	~$ curl \ 'https://example.hx.spiderfoot.net/api?func=scanresults&apikey=YOUR_API_KEY&id=bd499640df058efe3bfeb5e599c34cd0bc00cc49a37b66c4fa69b698b15bb781&type=IP_ADDRESS&paged=1&start=0&end=100' [{"fp": 0, "confidence": 100, "risk": 0, "parent_fp": 0, "notes": null, "parent_notes": null, "visibility": 100, "source": "binarypool.com", "date_found": "2015-09-27 08:17:08", "module": "sfp_dns", "data": "104.236.67.238", "id": "c202875808c4a99f8d04851df4507520f0ce15f7aee2880b5ce907f6da7fcbac"}, {"fp": 0, "confidence": 100, "risk": 0, "parent_fp": 0, "notes": null, "parent_notes": null, "visibility": 100, "source": "www.binarypool.com", "date_found": "2015-09-27 08:16:16", "module": "sfp_dns", "data": "104.236.67.238", "id": "1c9a0932b44055c4b4dfd56833b100f16621f96ed25d7483a3f012a68742f3ed"}

Name	scanresultsearch
Description	Searches a scan's result set for a string or regular expression, returning the results.
Arguments	*id* - The ID of the scan.
	value - The search string. Can be a simple wildcard search in the form of **keyword** or a regular expression in the form of /regex/ (e.g. /.*keyword.*/)
Returns	An array of hashes for each data element with the following fields: - *id*: Internal ID of this data element. - *source*: The source data from which this data element was found. - *date_found*: Date the data element was found. - *data*: The actual data itself, e.g. the IP address, hostname, etc. - *module*: Module that generated this element - *confidence*: Confidence-score (not yet used) - *visibility*: Visibility-score (not yet used) - *risk*: Risk-score (not yet used) - *fp*: Flag (0 or 1) indicating whether this data element has been flagged as a false positive by the user. - *parent_fp*: Flag (0 or 1) indicating whether the data element that generated this data element was set as a false positive by the user. - *notes*: User-supplied notes about this data element. - *parent_notes*: User-supplied notes about the data element that generated this data element.

	NOTE: The maximum number of results returned by a search is 1000.
Example	~$ curl \ 'https://example.hx.spiderfoot.net/api?func=scanresultsearch&apikey=YOUR_API_KEY&id=bd499640df058efe3bfeb5e599c34cd0bc00cc49a37b66c4fa69b698b15bb781&value=*104.236*' [{"fp": 0, "confidence": 100, "risk": 0, "parent_fp": 0, "notes": null, "parent_notes": null, "visibility": 100, "source": "www.binarypool.com", "date_found": "2015-09-27 08:16:16", "module": "sfp_dns", "data": "104.236.67.238", "id": "1c9a0932b44055c4b4dfd56833b100f16621f96ed25d7483a3f012a68742f3ed"}, {"fp": 0, "confidence": 100, "risk": 0, "parent_fp": 0, "notes": null, "parent_notes": null, "visibility": 100, "source": "binarypool.com", "date_found": "2015-09-27 08:17:08", "module": "sfp_dns", "data": "104.236.67.238", "id": "c202875808c4a99f8d04851df4507520f0ce15f7aee2880b5ce907f6da7fcbac"}]

Schedules

Name	schedulelist
Description	Lists all schedules that are in your SpiderFoot instance.
Arguments	*None required.*
Returns	An array of hashes for each schedule with the following fields: - *schedule_id:* The ID number of the schedule. - *schedule_name:* The name given to the schedule. - *scan_target:* The target being scanned. - *modules:* The modules enabled for the scan. - *frequency:* The frequency of the scan (d = daily, w = weekly, m = monthly) - *active:* Whether the scan is active (1) or disabled (0). - *last_run:* The date and time when the last scan was initiated via this schedule. Note that if the first scan was run via the UI as part of the schedule creation, the date and time will reflect that point in time, even though the scan was not actually run from the scheduler.
Example	~$ curl \ 'https://example.hx.spiderfoot.net/api?func=schedulelist&apikey=YOUR_API_KEY' [{"scan_target": "binarypool.com", "modules": "sfp__stor_db,sfp_accounts,sfp_adblock,sfp_bingsearch,sfp_blacklist,sfp_coderepo,sfp_cookie,sfp_crossref,sfp_darksearch,sfp_defaced,sfp_dns,sfp_duckduckgo,sfp_email,sfp_errors,sfp_filemeta,sfp_geoip,sfp_googlesearch,sfp_historic,sfp_honeypot,sfp_hosting,sfp_intfiles,sfp_ir,sfp_junkfiles,sfp_malcheck,sfp_names,sfp_pageinfo,sfp_pastes,sfp_pgp,sfp_portscan_tcp,sfp_pwned,sfp_sharedip,sfp_shodan,sfp_similar,sfp_social,sfp_socialprofiles,sfp_spider,sfp_sslcert,sfp_strangeheaders,sfp_tldsearch,sfp_virustotal,sfp_vuln,sfp_webframework,sfp_websvr,sfp_whois,sfp_yahoosearch", "last_run": "2015-12-27 10:24:47", "schedule_id": 42, "frequency": "d", "active": 1, "schedule_name": "binarypool"}]

Changes

Name	changelist
Description	Lists all open (status of OPEN) changes that are in your SpiderFoot instance.
Arguments	*None required.*
Returns	An array of hashes for each change with the following fields: - *change_id:* The ID number of the change. - *created:* When the change was raised. - *description:* The description of the change. - *status:* The workflow status of the change, can be OPEN or FALSE POSITIVE - changes of status CLOSED are left out). - *notes:* Any notes that have been set for this change. - *data:* The data generated from the scan or previous scan, that led to this change. - *module:* The impacting module that generated the event, leading to this change. - *event:* The impacting event description for this change, e.g. IP Address. - *schedule_id:* The ID number of the schedule that generated this change. - *schedule_name:* The name of the schedule that generated this change. - *schedule_target:* The target that was scanned, resulting in this change being generated.

Example	`~$ curl \`
	`'https://example.hx.spiderfoot.net/api?func=changelist&apikey=YOUR_API_KEY'`
	`[{"status": "OPEN", "schedule_target": "binarypool.com", "description": "New Element Identified", "schedule_name": "binarypool", "created": "2015-12-27 23:07:43", "notes": null, "schedule_id": 42, "module": "sfp_tldsearch", "change_id": 261, "data": "binarypool.com", "event": "Similar Domain"}]`

Name	changesetstatus
Description	Sets the workflow status of a given change ID.
Arguments	*id* - ID of the change.
	status - The workflow status to set the change to. Can be OPEN, FALSE POSITIVE or CLOSED.
Returns	Tuple representing the success of the call made.

Example	~$ curl \ 'https://example.hx.spiderfoot.net/api?func=changesetstatus&id=3&status=CLOSED&apikey=YOUR_API_KEY' ["SUCCESS", ""]

Name	changeaddnotes
Description	Add notes to an change. This can be useful for when you might handle the change internally within your organisation and need to perhaps associate an internal reference number (e.g. a ticket number) into SpiderFoot HX for maintaining an audit trail of the changes being handled.
Arguments	*id* - The ID of the change.
	notes - Base64-encoded notes you would like to record. Caution: SpiderFoot does not maintain notes history, so if previous notes are set, they will be overwritten!
Returns	Tuple representing the success of the call made.
Example	~$ echo -n 'False Alarm' | openssl base64 -e RmFsc2UgQWxhcm0= ~$ curl \ 'https://example.hx.spiderfoot.net/api?func=changeaddnotes&id=3&notes=RmFsc2UgQWxhcm0=&apikey=YOUR_API_KEY' ["SUCCESS", ""]