Vision API Welcome Packet

Quick Links to the Tools mentioned in this packet:

API Developer Documentation Full Website
API Example Code
Examples of Query Formatting in the Search Cheat Sheet for the Search API

Vision API Endpoints

The Vision Application Programming Interface (API) is a RESTful web service that enables access to Vision data and data products:

- **SEARCH** endpoints allow for targeted, complex queries with various parameters and filters to retrieve full documents from Vision's indexed data collection.
- **SCORE** endpoints allow for requesting and retrieval of DARKINT score calculations and associated formula inputs. Scores are based on the quality, quantity, and recency of exposed data found in Vision's data collection.
- **ENTITY** endpoints allow for lookup of structured records associated with cryptocurrency, emails, IP addresses, or credit cards found in Vision's data collection.
- **RANSOMWARE** endpoint facilitates querying for organization mentions on a selection of Ransomware as a Service (RaaS) websites and blogs.

API Type	Endpoint Name	Path	Description
Search	Search	/api/v1/search	Form complex searches to query Vision's DARKINT data. Use various query parameters, filters, and options (full body, snippets, or metadata/non-body fields) to return documents that meet your criteria.
	Document	/api/v1/documents/{id}	Return an individual document from Vision.

Score	Submit	/api/v1/score/submit	Request an asynchronous DARKINT Score calculation.
	Status	/api/v1/score/status?id=	Check the status of a DARKINT Score calculation.
	Result	/api/v1/score/result?id=	Retrieve a DARKINT Score and its associated score formula inputs.
Entity	Credit Card Number	/api/v1/entity/ccn	Retrieve mentions of a single credit card number found anywhere in the DarkOwl Vision dataset.
	Cryptocurrency Address	/api/v1/entity/crypto-address	Retrieve mentions of a single cryptocurrency address found anywhere in the DarkOwl Vision dataset. Supported currencies: Bitcoin, Dash, Ethereum, Litecoin, Monero, ZCash.
	Email Address	/api/v1/entity/email-address	Retrieve mentions of a single email address found anywhere in the DarkOwl Vision dataset.
	IP Address	/api/v1/entity/ip-address	Retrieve mentions of a single ip address found anywhere in the DarkOwl Vision dataset. IPv4 and IPv6 addresses are supported.
Entity Set	Bank Identification Number	/api/v1/entity/bin	Retrieve all credit card numbers associated with a 6-digit bank identification number, with cvv and expiration date (if available), found anywhere in the DarkOwl Vision dataset.
	Email Domain	/api/v1/entity/email-domain	Retrieve all email addresses within a particular domain, with password and password type (if available), found anywhere in the DarkOwl Vision dataset.
Ransomware	Ransomware	/api/v1/ransomware	Query DarkOwl Vision’s DARKINT data collection for mentions of various organization attributes on ransomware sites.

Authentication and Authorization Headers

>> Visit this link to download example code to help connect to our API.

Our documentation is here: https://docs.api.darkowl.com, but we have included some best practices below on how to create a correctly formatted request to the Vision API.

Two headers must be present in each API request: (1) The current datetime, and (2) Your Authorization header. We'll go over how to format/calculate both of these.

1. The current datetime is formatted in UTC, and should look like this:
   Wed, 24 Oct 2019 16:59:00 GMT
2. Your Authorization Header will look something like this:
   OWL insertYourPublicKeyHere:ABCcEFgHIJKLmnOPqRStu+123/bbQ=

Step 1: Calculating Your Authorization Header

To create your Authorization Header, you will need the following:

- The current date (as formatted above)
- Your request verb in all caps, which is 'GET'
- The full path of your request after the host. As an example, the highlighted part of the full request below is the part that is needed:
  https://api.darkowl.com/api/v1/endpoint1?aParam1=val1&aParam2=val2

The procedure is as follows:

1. Concatenate the verb, full path, and date into a single string *(no spaces in between each, no newline character)*
   a. stringToSign = verb + full path + date

2. Run the StringToSign through the HMAC-SHA1 algorithm using your private key.

3. Base64 Encode the resulting HMAC hash value.

4. Add an 'Authorization' header value with the HMAC value and public key in the following format, noting the "OWL" keyword in the header as shown here:
   Authorization: OWL yourPublicKeyHere:resultFromStep3

>> Important notes on calculating your authorization header:

1. The HMAC function needs to return a raw binary result, not a hex string.
2. Calculate your HMAC before URL encoding the query string. DarkOwl will decode the encoded URLs prior to authenticating, so an HMAC ran on an encoded URL will result in a security hash mismatch.

>> Step 2 & Putting it all together are on the next page

Step 2: Adding the Date

Once your authorization header is completed, add a second header with the datetime, as shown here:

Date: Wed, 24 Oct 2019 16:59:00 GMT

Putting It All Together: The Request

To summarize, each request must have the following headers (*your exact values will vary):

Date: Wed, 24 Oct 2019 16:59:00 GMT

Authorization: OWL insertYourPublicKeyHere:resultFromStep3

Error Codes

We use typical HTTP response codes for bad requests and authentication issues. If your request generates an error, the response will included a more detailed message about the specific error. If you are unsure about what is causing the error, please contact us at productsupport@darkowl.com.

If you receive a 403 response code, please verify that your traffic is originating from an approved IP address for your organization.

Working with the Search Endpoint

>> Visit this link to download our Search Cheat Sheet for the Search API.

When working with the Search endpoint (/api/v1/search), the **'q' parameter** is the primary search field recommended for use with keywords or terms.

Each query parameter is designated as q (the Base Search), Filter, or Result Option:

- Note that *only the q parameter determines the relevancy score* of the documents that are returned by our database; filters are not used in the calculation of relevancy.
- Filters allow for more targeted, more performant searches, as they narrow down a result set.
- Result Options allow you to control the way documents are returned, such as with highlighting (highlight=), sorted (sort=), with pagination (offset=), etc.

The q field

The `q` parameter is the Base Search field and should be used with all searches. This field accepts letters, numbers, special characters, and operators. Wildcards are generally allowed, except for leading wildcards.

Using quotations and parentheses: Use quotations around multi-word phrases or names to group everything together as one item. Parentheses can be used to form subqueries.

Using Booleans: You can use AND, OR, NOT in this field, for example: hack AND breach.

Use search operators when searching for emails, cards, ssns, ips, cryptocurrency in the q field.

q=email:(first.last@company.com OR first@company.com)

q=cryptocurrency:griheoaho3249070

q=drugs AND email:(first.last@company.com OR first@company.com)

Exact Searching and stemming: DarkOwl Vision supports a process called stemming, which tries to reduce a word to an approximation of its stem or root form. This means that searching will return matches on related forms of a word (hope, hopes, hoped, and hoping may be returned in results when searching for hope), unless you specify otherwise. When you want to search for a specific term, including special characters and punctuation, use the exact: operator to prevent word stemming:

q=exact:hack

Using highlight with q: When you append your query with the highlight parameter (&highlight=true), data you enter in the q parameter will be highlighted in the body field in the response.

The q parameter determines relevancy: The relevancy score – how well the result matches the query submitted – is determined by the content in the q parameter; note that filters are not used in the calculation of relevancy.

Paginating through results (Offset parameter) and Count parameter

A maximum of **20 results are returned per request.** The 'offset' parameter allows you to skip a number of results. If your query has more than 20 total documents, use the following to get the 'next' page of results, with a maximum of 5,000* results returned for the same query.

- offset=0 will return the first 20 results; this is the default
- offset=20 will return results 21-40
- offset=40 will return results 41-60
- etc

*Please see our documentation for current maximum pagination and offset values.

When you're initially developing and debugging, you can ask the Search API to return fewer than the default 20 results. The count= parameter allows you to set a number between 1 and 20.

Note that when using a count other than the default (20), you may need to adjust your offset. For example, if you set your count to 2 (count=2) while developing, you would use offset=0 to return the first 2 results, offset=2 to return results 3-4, offset=4 to return results 5-6, and so on.

De-duplicating results (Similar parameter)

You can ask for your documents to be de-duplicated by the Search API, per request. Note that if you select this option, you may receive fewer documents than the count selected, since some of the results may not be returned. You will receive the number of results actually sent in the resultCount in the response.

Leak Descriptions (Leak parameter)

The leak parameter can be used to filter your search to (or exclude from) known public leaks or database dumps. The parameter values of current leaks available through Search API are found here.

Description of Result Fields

Response fields	**Field descriptions**
id	The DarkOwl Vision identifier for the result.
body	The full text collected from the webpage/record/target. Note that this field will not be returned if *detail=snippet* or *detail=nonbody* is selected in the request.
snippet	Excerpt of the body, based on the first highlighted term in the body. This field will not be returned if *detail=body* or *detail=nonbody* is selected in the request, or if the request does not include a detail parameter (the detail default is body).
hackishness	Rating assigned by DarkOwl Vision, indicating the likelihood to which the information could be used for criminal activity.
title	If available, page title of the content collected.
url	URL or location of the content collected.
crawlDate	Date when DarkOwl Vision collected the content.
fileSize	The size of the content before normalization, in bytes.
domain	Domain of the content collected.
ips	A list of ip addresses found in the body, if available.
emails	A list of emails found in the body, if available.
ssns	A list of social security numbers found in the body, if available.
ccns	A list of credit card numbers found in the body, if available.
cryptos	A list of cryptocurrencies found in the body, if available.
headers	The httpHeader content collected with the result, if available.
leak	Leak information and metadata, if the document was sourced from a leak. May include the following fields, if available: • name • actors • host • associations • downloadLocations • filepath, filename

Working with the Score Endpoints

>> Visit this link to download Python 3 example code that can help make a DARKINT score submission and retrieval.

The Submit, Status, and Result endpoints work in tandem to perform an end-to-end score calculation and retrieval. The DARKINT Score formula focuses on specific DARKINT sources for unique matches on an organization's website and email domains, and adjusts the results based on hackishness. Inputs include one or more domain(s) and emailDomain(s).

The result includes the score, the number of document matches for the domain and emailDomain inputs provided, and the calculated hackishness values.

Response fields	**Field descriptions**
score	Calculated score, based on the DARKINT Exposure Score formula (see next section).
domainPaste	Number of document matches from Paste sources that include domain input value(s).
domainDark	Number of document matches on Darknet sources that include domain input value(s).
domainBreach	Number of document matches within Data Leaks that include domain input value(s).
emailPaste	Number of document matches on Paste sources that include email input value(s).
emailDark	Number of document matches on Darknet sources that include email input value(s).
emailBreach	Number of document matches within Data Leaks that include email input value(s).
hackishness DarkPaste	Average hackishness value of document matches on Paste and Darknet sources that include domain or email input value(s), occurring within the last 90 days.
hackishness Breach	Average hackishness value of document matches within Data Leaks that include domain or email input value(s), over all time.

How the Score is Calculated

$$\textbf{DARKINT SCORE} = H_{90} * (\ln RDS + \ln RTS) + H_{ATR} * (\ln ATR)$$

H_{90} = Hackishness of last 90 days results

H_{ATR} = Hackishness of all time Data Leak results

RDS = # results from Darknet Sites

RTS = # results from Transitory Sites

ATR = # results from all time Data Leak results

Working with the Entity Endpoints

The Entity endpoints allow you to retrieve structured records associated with tokenized values discovered in our DARKINT data collection.

- There are four (4) Entity endpoints that allow you to look up and return records related to an individual or singular **credit card number, cryptocurrency address, email address,** or **IP address.**
- There are two (2) Entity endpoints that allow you to look up and return records related to a set of email addresses (from one **email domain**) or a set of credit cards (from one **bank identification number**).

Supported Entity Inputs

Cryptocurrency Address	Bitcoin, Dash, Ethereum, Litecoin, Monero, ZCash are supported.
IP Address	IPv4 and IPv6 addresses are supported.
Bank Identification Number	6- or 8-digit Bank Information Numbers are supported.

Entity Request Options

Entity types have several common request options, including sort and date range (from/to) options.

Certain Entity types may also have unique request parameters, as shown in the table below.

Entity	**Unique Request Parameter**	**Description**
Email Domain	*leak*	Use this parameter to only return content from one specific data leak.

Entity Response Fields

All Entity types have common response options, including id, crawlDate, location, fragment, and network.

Certain Entity types may also return unique options, as shown in the table below. If these fields are not returned, it means that field was not detected with the Entity.

Entity	**Unique Response**	**Description**
Email Address	*password*	An associated password that Vision detected with the email address.
Email Address	*type*	The type of password that Vision detected with the email address (plain, hashed).

Email Domain	*password*	An associated password that Vision detected with the email address.
Email Domain	*type*	The type of password that Vision detected with the email address (plain, hashed).
Email Domain	*leak*	The name of the data leak in which the result was found (if from data leak).
Credit Card Number	*cvv*	An associated cvv that Vision detected with the credit card number.
Credit Card Number	*expDate*	An associated expiration date that Vision detected with the credit card number.
Bank Identification Number	*cvv*	An associated cvv that Vision detected with the credit card number.
Bank Identification Number	*expDate*	An associated expiration date that Vision detected with the credit card number.

Working with the Ransomware Endpoint

The Ransomware endpoint allows you query for organization mentions or other attributes, filtered to a selection of Ransomware as a Service (RaaS) websites and blogs.

Base Search Fields

At least one Base Search field (see table below) must be used with all searches.

- Base search fields accept letters, numbers, special characters, and operators. Wildcards are generally allowed, except for leading wildcards.
- Up to 10 Base Search fields can be used in a single query.

Base Search Fields	**Example**	**Description**
org_name	*Mega Corp*	Use to look for mentions of an organization's name.
org_domain	*megacorp.net*	Use to look for mentions of an organization's domain or URLs containing the organization's domain. Be sure to specify the domain portion only (i.e. megacorp.com) and not a full URL (i.e. https://megacorp[dot]com).
contact_name	*Cecilia Young*	A first and last name. Use to look for mentions of a CEO or key executives mentioned on ransomware sites.
keyword	*bitcoin*	This field supports either a word or a phrase. Phrases do not need to be in quotations.

Paginating through results (Offset parameter) and Count parameter

A maximum of **20 results are returned per request.** The 'offset' parameter allows you to skip a number of results. If your query has more than 20 total documents, use the following to get the 'next' page of results, with a maximum of 5,000* results returned for the same query.

- offset=0 will return the first 20 results; this is the default
- offset=20 will return results 21-40
- offset=40 will return results 41-60
- etc

*Please see our documentation for current maximum pagination and offset values.

When you're initially developing and debugging, you can ask the Ransomware API to return fewer than the default 20 results. The count= parameter allows you to set a number between 1 and 20.

Note that when using a count other than the default (20), you may need to adjust your offset. For example, if you set your count to 2 (count=2) while developing, you would use offset=0 to return the first 2 results, offset=2 to return results 3-4, offset=4 to return results 5-6, and so on.

De-duplicating results (Similar parameter)

You can ask for your documents to be de-duplicated by the Ransomware API, per request. Note that if you select this option, you may receive fewer documents than the count selected, since some of the results may not be returned. You will receive the number of results actually sent in the resultCount in the response.

Description of Result Fields

Response fields	**Field descriptions**
id	The DarkOwl Vision identifier for the result.
body	The full text collected from the webpage/record/target. Note that this field will not be returned if *detail=snippet* or *detail=nonbody* is selected in the request.
snippet	Excerpt of the body, based on the first highlighted term in the body. This field will not be returned if *detail=body* or *detail=nonbody* is selected in the request, or if the request does not include a detail parameter (the detail default is body).
hackishness	Rating assigned by DarkOwl Vision, indicating the likelihood to which the information could be used for criminal activity.
title	If available, page title of the content collected.
url	URL or location of the content collected.
crawlDate	Date when DarkOwl Vision collected the content.

fileSize	The size of the content before normalization, in bytes.
domain	Domain of the content collected.
ips	A list of ip addresses found in the body, if available.
emails	A list of emails found in the body, if available.
ssns	A list of social security numbers found in the body, if available.
ccns	A list of credit card numbers found in the body, if available.
cryptos	A list of cryptocurrencies found in the body, if available.
headers	The httpHeader content collected with the result, if available.