Vision API Welcome Packet

Quick Links mentioned in this Welcome Packet:

API (Developer) Documentation || Example Code || API Values || Additional Resources

Vision API Endpoints

The Vision Application Programming Interface (API) enables access to Vision data, data enrichment, and analytical products:

Data: Indexed Collection

- **SEARCH** endpoint allows for targeted, complex queries with various parameters and filters to retrieve full documents from the entirety of Vision's indexed data collection.
- **RANSOMWARE** endpoint facilitates querying for organization information specifically filtered to Ransomware as a Service (RaaS) websites and blogs, or leaks sourced from those RaaS sites, from Vision's indexed data collection.
- **DOCUMENT** endpoint allows for retrieval of specific documents from Vision's indexed data collection.

Data: Structured Records

- **ENTITY** endpoints allow for lookup of structured records associated with cryptocurrency, emails, IP addresses, or credit cards.

Data Enrichment

- **LEAK** endpoints allows you to retrieve a list of leaks recently added to the DarkOwl Vision dataset, or supplemental information about an individual leak.

Analytical Products

- **ACTOR** endpoints provide actor dossiers for selected threat actors, which may include profile information, targets, tools, and/or darknet fingerprint.
- **DARKSONAR** endpoint allows for retrieval of a relative risk rating for a domain, which captures rising or falling cyber risk over time. Ratings are based on credential exposure of a domain as seen in Vision's data collection, compared with the domain's baseline.
- **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.

API Type	Endpoint	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). Documents that meet your criteria are returned with all metadata fields; full body or snippet detail options are available.
	Document	/api/v1/documents/{id}	Return an individual document from Vision, including all metadata fields.

API Type	Endpoint	Path	Description
Ransomware	Ransomware	/api/v1/ransomware	Query DarkOwl Vision’s DARKINT data collection for mentions of various organization attributes on ransomware sites or leaks sourced from ransomware sites.
Entity	Credit Card Number	/api/v1/entity/ccn	Retrieve mentions of a single credit card number found in the DarkOwl Vision dataset.
	Crypto-currency Address	/api/v1/entity/crypto-address	Retrieve mentions of a single cryptocurrency address found 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 in the DarkOwl Vision dataset.
	IP Address	/api/v1/entity/ip-address	Retrieve mentions of a single ip address found in the DarkOwl Vision dataset. IPv4 and IPv6 addresses are supported.
	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 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 in the DarkOwl Vision dataset.
Leak	Leak Context	/api/v1/context/leak	Retrieve information about a data leak in the DarkOwl Vision dataset.
	Leak Summary	/api/v1/context/summary	Retrieve a list of data leaks that were recently added into the DarkOwl Vision dataset.
Actor	Actor	/api/v1/actor	Retrieve an actor dossier for a selected threat actor.
	Summary	/api/v1/actorsummary	Retrieve a list of actor names within the DarkOwl Actor dataset.
DarkSonar	DarkSonar	/api/v1/risk	Request and retrieve a synchronous relative risk rating and signal interpretation. Includes an option to include historical ratings and baseline calculations in the response.
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.

Authentication and Authorization Headers

Example Code: Visit this Link

API Documentation Section: Authentication

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 either 'GET' or 'POST' depending on the endpoint
- 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, *including the / after .com*:

  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.

(continues on next page)

3. Base64 Encode the resulting HMAC hash value.
    a. The HMAC function needs to return a raw binary result, not a hex string.
    b. **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.
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

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 include a more detailed message about the specific error. If you are unsure about what is causing the error, please contact us at support@darkowl.com.

- **If you receive a 403 response** code, please verify that your traffic is originating from an approved IP address for your organization.
- **If you receive a 401 response** code, please verify that your authentication header is correct.
    - One common thing to check: when you base64 encode the output of your HMAC calculation, be sure that it's base64 encoding the raw binary result, not a hex encoded string of the HMAC output.

Working with the Search Endpoint

Example Code: Visit this Link

API Documentation Section: Search

Additional Links: Search Cheat Sheet, UI to Search API Field Map, API Values

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 either **q (the Base Search)**, **Filter**, or **Result Option**:

- *Only the q parameter determines the relevancy score* of the documents returned by our database; filters are not used in the calculation of relevancy.
- Filters allow for more targeted, 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.

Base Search Field (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.
- **Using Search Operators for Entities:** 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. *Please see the API documenation for current maximum pagination and offset values.*

- 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

When you're initially developing and debugging, you can ask the Search API to return fewer than the default 20 results. The *count* parameter accepts numbers between 1 and 20.

- ***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, on a per request basis.

- **When this option is selected, you may receive fewer documents** than the count submitted, since some of the results may not be returned. You will receive the number of results actually sent in the *resultCount* in the response.

Leaks available in Vision (Leak parameter)

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

Description of Response Fields in Search API

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. Assigned on a per-page basis.
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.
cves	A list of Common Vulnerabilities and Exposures 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* name: The name DarkOwl assigned to the leak, which is typically based on the target, original post title or file name, or other generally-known name. **To retrieve Leak Context:** Pass the value from this *name* field to the Leak Context endpoint. actors: The name of the original poster or actor who made the data available, if known. host: The site name of the location where the leak was hosted. Examples: Doxbin, BreachForums, Mega. associations: Other categories that help classify the leak, or associate it with other leaks. Examples: Combolist, Ransomware, Twitter (if there are multiple Twitter-related breaches).
chat	Chat information and metadata, if the document was sourced from a chat network. May include the following fields, if available: *users, channel, server*
forum	Forum post information and metadata, if the document was sourced from a forum. Will include the following fields: *threadId, postAuthor, postDate.*

Working with the Ransomware Endpoint

API Documentation Section: Ransomware

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. *Please see the API documenation for current maximum pagination and offset values.*

- 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

When you're initially developing and debugging, you can ask the Ransomware API to return fewer than the default 20 results. The *count* parameter accepts numbers between 1 and 20.

- ***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.

Response Fields in Ransomware API

The response fields are the same as Search API. Please refer to the table on page 9 for detailed descriptions of each field.

Working with the Entity Endpoints

Example Code: Visit this Link

API Documentation Section: Entity, Entity Set

Additional Links: API Values

The Entity endpoints allow you to retrieve structured records relating to four tokenized values identified in our data collection—*emails, IP addresses, cryptocurrency, or credit cards.*

- 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 (belonging to one **email domain**) or a set of credit cards (belonging to one **bank identification number (BIN)**).

Supported Entity Inputs

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

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 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.

Entity	Unique Response	Description
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 Leak Endpoints

API Documentation Section: Leak Context, Leak Summary

Leak Context allows you to retrieve supplemental information about data leaks in the DarkOwl Vision dataset. **Leak Summary** allows you to retrieve a list of data leaks that were recently added into the DarkOwl Vision dataset.

Leak Context Base Search Field (name parameter)

The `name` parameter is the Base Search field for Leak Context. This is the name of the leak for which you are retrieving supplemental information. This value for this field is the value returned in the *name* field in a Search API leak result.

Leak Context Response Fields

Note: Individual leaks may have varying amounts of information returned, depending on what is known about that leak. Not all fields may be returned for all leaks; additionally, the list of returned fields is subject to change, as DarkOwl may add new returned fields when additional information becomes available.

Response fields	**Field descriptions**
Name	Name of the leak.
Description	A short description about the nature of the leak.
Date Available	Date the data was made available on the darknet or internet, if known.
Date Breached	Date of the leak incident or ransomware attack, if known.
Content Categories	High-level categories or types of content contained in this data leak. Content Category examples include: combolist, credentials, documents, messages, PII, stealerlogs, etc.
Content Specifics	More granular information about the compromised data in this leak. Content Specifics examples include: dates of birth, email addresses, financial documents, internal documents, ip addresses, legal documents, plaintext passwords, phone numbers, physical addresses, profile information, usernames, etc.
Password Format	Format of passwords found in the leak, if applicable, such as: plaintext, hashed, none, both.
Password Hash Formats	If the leak contains hashed passwords, this field will display what hashing algorithm is used, if known. Examples: MD5, SHA1, etc.
Associations	Any entity (site, organization, country, year, etc) that is associated with the data leak. Values in this field are searchable with the leak: operator.
Actors	Username(s) of the original poster or actor responsible or otherwise involved in leaking the data. Values in this field are searchable with the leak: operator.
Attack Types	The type of attack that resulted in the data leak.
Targets	The target organization or company where the data originated, if known. This is generally the name of the organization(s) attacked. If known, this field will optionally return Target Name, Target Domain, Target Description, and/or Target Country.
Countries	Country associated with the leak; for leaks related to a country without an organization Target.

Response fields	Field descriptions
Hosts	Site name(s) on which the original data was hosted. Values in this field are searchable with the leak: operator.
Download Locations	The URL where the leak was downloaded.
Original Post URL	The URL of the post in which the leak was initially shared.
Original Telegram Channel	The Telegram Channel ID in which the leak was initial shared. This value can be passed in Search API's \`telegramChannel\` parameter for more research.
Post Location Type	A classification of the location on which the leak was initially shared. Examples include: forum, leak site, marketplace, messaging platform, telegram, torrent, etc.
Post Vision ID	The document ID of the original post in the DarkOwl Vision index, if known.
Associated URLs	Any additional URLs that may be associated with this leak.
Leak Classifications	A classification of the nature of the leak. Examples include: Combolist, Cyberwar, Politically Motivated, Ransomware, Stealerlogs, etc.
Leak Size Records	The total number of records contained in the leak.
Leak Size Actual	The actual size of the leak, once downloaded.
Leak Size Advertised	The advertised size of the leak, from the original post.
Completeness	In some cases, partial or sample data is leaked by an actor. This field will display *Partial* if it is known that the leak dataset is not complete.
For Sale?	In some cases, leak data is offered for sale prior to being released on the darknet by an actor. This field indicates whether the leak content was ever offered for sale, if known. *Note: DarkOwl adheres to a strict collections policy guided by CCIPS best practices, and we do not purchase data or facilitate criminal activity.*
Filetree	The name of the filetree document in DarkOwl Vision. Values in this field are searchable with the leak: operator.
Public Reporting	Any URLs, dates, or notes related to public reporting about this leak.
Media Reporting	Any URLs, dates, or notes related to media reporting about this leak.

Working with the Actor Endpoints

API Documentation Section: Actor, Actor Summary

The **Actor API** allows you to retrieve actor dossiers for selected threat actors, which may include profile information, targets, tools, and/or darknet fingerprint. **Actor Summary** allows you to retrieve a list of available actor names within DarkOwl's curated actor database.

Actor Base Search Field (name parameter)

The `name` parameter is the Base Search field for Actor API. This is the name of the actor for which you are retrieving supplemental information. This value for this field is any value returned in the *name* field from the Actor Summary response.

Actor Response Fields

Note: Individual actors may have varying amounts of information returned, depending on what is known about that actor. Not all fields may be returned for all actors; additionally, the list of returned fields is subject to change, as DarkOwl may add new returned fields when additional information becomes available.

Field name	Description
name	Name of the threat actor.
dateUpdated	Date at which this entry was most recently updated.
dateFirstSeen	Date at which this threat actor was first seen.
dateLastSeen	Date at which this threat actor was last seen.
aliases	List of known aliases for this threat actor.
actorSize	Size of the threat actor. Options include: group, individual, unknown.
countries	Country or countries from which the actor is known to operate.
actorTypes	Descriptors for the type of threat actor group. Options include: activist, competitor, crime-syndicate, criminal, hacker, insider-disgruntled, nation-state, sensationalist, spy, terrorist.
leakNames	List of leak names with which the threat actor is known to be associated.
marketNames	List of market names on which the threat actor is known to be associated with/active.
forumNames	List of forum names on which the threat actor is known to be associated with/active.
forumMarket Usernames	List of usernames the actor is known to use on forums and/or markets.
keyPoints	Description/main points about the threat actor.
lawEnforcement	Description of any law enforcement activity related to the threat actor (if known).
contactInfo	Any known contact information for the threat actor. Individual objects within this field will include the type of contact and an array of strings for relevant addresses of that type.
paymentInfo	Any known payment information for the threat actor. Some types include: Bitcoin, Monero, PayPal. Individual objects within this field will include the type of payment and an array of strings for relevant addresses of that type.
websites	List of website domains that the threat actor is known to be associated with/active on.

Field name	Description
socialMediaInfo	Any known social media handles for the threat actor. Individual objects within this field will include the type of payment and an array of strings for relevant addresses of that type.
cves	List of CVEs that the threat actor is known to be associated with/has exploited.
tools	List of tools that the threat actor is known to be associated with/has used.
actorSophistication	Description of the actor's relative sophistication level, using STIX classification. Options in increasing order of sophistication: unknown, minimal, intermediate, advanced, expert, innovator, strategic.
actorSpecializations	Area(s) in which the threat actor is known to specialize. For example: ransomware, espionage, exploit developer, etc.
targets	Any known targets of the threat actor. Individual objects within this field will include the category of the target and the name of the target.
isActive	Is the actor known to be currently active.

Working with the DarkSonar Endpoint

Example Code: Visit this Link

API Documentation Section: DarkSonar

Additional Links: Cyber Risk Modeling: Introducing DarkSonar

DarkSonar is a risk metric based on darknet intelligence and measures an organization's credential exposure on the darknet. It is a relative risk rating that considers the nature, extent and severity of credential leakage on the darknet to provide a company with a signal that acts as a measurement for a company's exposure.

The DarkSonar endpoint allows you to retrieve a relative risk rating for a domain, which captures rising or falling cyber risk over time. The rating is based on email exposure using three parts of email entities: *unique plaintext credentials, unique hashed credentials,* and *total unique email address volume with no credentials*.

DarkSonar Request Options

When working with the DarkSonar endpoint, the ***domain*** parameter is the primary field used, along with the ***detail*** parameter.

- *detail=rating* will return the current rating (numerical value) and its risk interpretation signal (low, elevated, severe).
- *detail=full* will return the above, plus an array of historical monthly baseline details for the past 24 months. These baseline details are the underlying values upon which the current rating and signal are determined. We recommend graphing these values on a timeline for visualizing purposes.

DarkSonar Response Fields

Response fields	**Field descriptions**
rating	The z-score for the domain, which is calculated based on the past 24 months of baseline ratings for the domain. The z-score indicates the number of standard deviations away from the mean at this point in time. A DarkSonar rating that is trending upwards away from the mean, or over 1 standard deviation from the mean, can be an early warning sign of risk. A null rating is returned in the instance that the domain has zero results in the Vision dataset.
signal	Interpretation of the rating, based on its proximity to the baseline rating. *Note these are suggested interpretations and are based on the number of standard deviations away from the mean.* • Ratings lower than 1 will be considered low. • Ratings equal to or higher than 1 but less than 2 will be considered elevated. • Ratings equal to or higher than 2 will be considered severe.

Response fields	Field descriptions
date	Date the current rating and signal were calculated.
details	This is a list of objectives outlining monthly historical numbers for this domain, over the past 24 months. It will be an array that includes: • ***date:*** date at which this baseline and z-score rating were calculated. • ***baseline:*** the baseline rating, which is measured by credential exposure on the darknet, with considerations for age and uniqueness. • ***rating:*** The z-score for the domain, which is calculated based on the past 24 months of baseline ratings for the domain. The z-score indicates the number of standard deviations away from the mean at this point in time.

Working with the Score Endpoints

Example Code: Visit this Link

API Documentation Section: Score

Additional Links: Guide with Score Formula

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).

Score Response Fields

The response 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.
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.