APNIC Stats API Examples

Production Endpoint: <https://api.apnic.net:8280/stats/1.0.0/>

Please see the API console for latest list of accepted methods, request parameters and parameter values.

# Authentication & Headers (required)

All your requests must be accompanied by a valid Authorization Header and access token – if you do not, you will get an error message “Missing Credentials”.

When calling the API, send the values as the OAuth 2.0 access token with authentication type set as Bearer.

Example: Authorization: Bearer {accessToken}

Additionally, set the Accept header to application/json

# GET Request Method

Request without parameters:

GET /stats/1.0.0/

cURL request:

curl –H “Authorization: Bearer {access code}” –X GET –v https://api.apnic.net:8280/stats/1.0.0/

Response:

1. HTTP/1.1 200 OK
2. Date: Thu, 12 Oct 2010 20:13:58 GMT
3. Last-Modified: Tue, 18 Feb 2014 12:39:02 +1000
4. Content-Type: application/json

{

"request": {

"economy": "AU,NZ,JP",

"action": "stats",

"limit": 100,

"offset": 0,

"apiVersion": 1

},

"results": [

{

"economy": "AU",

"resource": "ipv4",

"address": "1.0.0.0",

"allocation\_length": 256,

"date": "20110811",

"status": "assigned"

},

{

"economy": "JP",

"resource": "ipv4",

"address": "1.0.16.0",

"allocation\_length": 4096,

"date": "20110412",

"status": "allocated"

},

…

}],

"serverInformation": {}

}

## Adding request parameters

You can append a single parameter to the endpoint URL by adding a question mark (?), the name of the parameter filter, an equals sign (=) and the value you wish to filter on.

Format:

{parameter-name}={parameter-value}

Example 1: “<endpoint>/?economy=AU” will return all results matching economy ‘AU’ (Australia).

Request with single parameter:

GET /stats/1.0.0/?economy=AU

## Request parameter with multiple values

You can specify multiple values the filter by comma separating the requested values. Ensure there aren’t any trailing commas. There is no limit to the number of parameter values you are allowed to specify.

Format:

{parameter-name}={parameter-value1},{parameter-value2},{parameter-value3}

Example 2: “<endpoint>/?resource=ipv4,ipv6” will return all results matching resource type ‘ipv4’ or ‘ipv6’. No ASN results will be returned.

Request with single parameter with multiple values:

GET /stats/1.0.0/?resource=ipv4,ipv6

Response:

1. HTTP/1.1 200 OK
2. Date: Thu, 12 Oct 2010 20:13:58 GMT
3. Last-Modified: Tue, 18 Feb 2014 12:39:02 +1000
4. Content-Type: application/json
5. {
6. request: { },
7. results: { },
8. serverInfomation: { }
9. }

## Multiple request parameters

You can add additional request parameters to the URL by appending them to the single parameter request URL, separated with an ampersand (&). Only one for each parameter type

Format:

{parameter-name1}={parameter-value1}**&**{parameter-name2}={parameter-value2}…

Example 3: “<endpoint>/?economy=AU,NZ,JP,TW&resource=ipv4,ipv6” will return all results matching economy code AU, NZ, JP or TW and resource type ‘ipv4’ or ‘ipv6’.

## Sorting

Data is sorted by default by the address in ascending order. However, you are able to specify the sort column and also the sort direction of the data. The “sortby” and “sortdir” can be used independently or together.

Valid values for the “sortby” are any names of the fields returned in the data set. E.g. “economy”, “resource”, “address”, “allocation\_length”, “date”, “status”.

“sortdir” can be either “ASC” for ascending or “DESC” for descending.

Format:

sortby={sort-by-name}**&**sortdir={ASC or DESC}

## Limits

By default the 100 results will be returned, however you can request less or more results by specifying the number with a limit parameter. It can be used in conjunction with your other filter parameters.

Valid values for the ”limit” parameter are any positive integer, or for the complete dataset (no limit), you can enter “unlimited” or “-1”.

Format:

limit={number-of-results-to-return}

## Offsets

By default an offset of 0 is used when none is set. You can use offsets to paginate through the data and are useful if you wish to limit the amount of data returned from a single request but still wish to obtain the full set of data. If you specify an offset greater than the total number of results in the data set, it will return no results.

If the data set (based on filters) contains 250 results and you are fetching 100 results at a time, you can get the whole data set in 3 requests. (i.e. offset=0, offset=100, offset=200 – the last request will return 50 results)

Format:

offset={number-of-results-to-skip}

## Documents

The public API provides access to two datasets. There is all the “LATEST” data available from all the registries and the “APNIC” historical allocation data. The default behaviour is to search over both data sets “ALL”.

Format:

documents={LATEST or APNIC or ALL}

## Date Range

The date range parameter allows you to select results from the APNIC historical data. The date is the file date. Currently you can only supply one date per query.

Format:

daterange={YYYYMMDD}

## Using the Group-By Parameter

You can use the group-by parameter to return counts of your data rather than the individual records. This can be useful for charting or if you simply wish to get an overview of the data.

Grouping can be performed on each of the parameter options. E.g. “economy, status, resource”.

IPv4 /8s counts are returned in “ipv4\_count”, IPv6 /32s counts are returned in “ipv6\_count” and ASN counts are returned in “asn\_count”.

You can also specify filters for the groupby queries, and counts will be based on data returned after the filter has been applied. The “sortby”, “sortdir”, “limit” and “offset” parameter filters are not valid groupby options and do not work as filters on the data that is grouped.

Format:

groupby={group-by-name}

Example 4: “<endpoint>/?groupby=economy&economy=AU,NZ,JP” will return counts for grouped with the economy codes used as keys object count keys, only for AU, NZ and JP.

Response:

1. HTTP/1.1 200 OK
2. Date: Thu, 12 Oct 2010 20:13:58 GMT
3. Last-Modified: Tue, 18 Feb 2014 12:39:02 +1000
4. Content-Type: application/json
5. {
6. request: {…},
7. results: { {
8. "AU": {
9. "ipv4\_count": 2.8477783203125,
10. "ipv6\_count": 8659.011825561523,
11. "asn\_count": 1591
12. },
13. "JP": {
14. "ipv4\_count": 12.016067504882812,
15. "ipv6\_count": 11251.126693725819,
16. "asn\_count": 1048
17. },
18. "NZ": {
19. "ipv4\_count": 0.41534423828125,
20. "ipv6\_count": 156.0010528564453,
21. "asn\_count": 422
22. }
23. }
24. },
25. serverInformation: {…},
26. }

# Method: POST

As an alternative to using GET, you can submit POST data requests to the API endpoint. So instead of appending your request parameters to the URL, you can instead submit them in a JSON object.

The same parameters (filters) are accepted and will return same results as you would get from using a GET request. Ensure it is a properly formed JSON object (no trailing commas), and that the parameter values are passed as strings, with the exception of the limit and offset parameters, which you can pass as integers.

JSON request object format:

{

{request-parameter-name}: “{request-parameter-value}”

}

Or with multiple parameter filters:

{

{request-parameter-name1}: “{request-parameter-value1},

{request-parameter-name2}: “{request-parameter-value2},

{request-parameter-name3}: “{request-parameter-value3}

}

cURL request:

curl –H “Authorization: Bearer {access code}” –H “Content-Type: application/json” –X POST –v <https://api.apnic.net:8280/stats/1.0.0/> -d “{

‘economy’: ‘AU,NZ,JP’,

‘status’: ‘allocated’,

‘resource’: ‘ipv4,ipv6’,

‘limit’: ‘300’

}”