Acled API

User Guide

Richard Holmes

https://ampersandstudio.uk/

February 2018

Contents

Acled API

Version 2.0

Introduction

The following document highlights the basic steps for interacting with the Acled API. The API is RESTful in nature and is accessed via Basic HTTP(S) authentication.

API Access Detail

The full URL for accessing the API is https://api.acleddata.com/{data}/{command}, where "data" represents the type of data to be collected and "command" represents the action to be executed. See below for details regarding possible data types and usages.

Sample API Calls and Responses

API calls may be made in any standard browser or using any programmatic language that is capable of making HTTP requests. The samples below demonstrate the URL to be called to make the request.

The following points should be noted:

- This API only uses the GET or POST request. A GET request is advised whereby all data sent will be contained within standard Query String parameter formats and URLencoded.
- All responses from the API shall be formatted as JSON unless specifically requested in either XML, CSV or TXT format.
- TXT format returns a plain text csv string.
- A limit of 500 lines of data (1000 rows for monadic) will be returned by default for Acled and Actor data types. Larger requests may be made, however.
- All fields, specific to the data type,will be returned by default. Reduced field lists can be requested for some data types.
- Acled data is returned in date order DESC (starting with the latest).

Response Format

By default the response is returned in JSON format but it's possible to return the response in XML, CSV and TXT format too. In order to return another format you simply add the

relevant extension to the end of the command name so the query would look like the following:

Format	HTTP Request Format	MIME Type
JSON	https://api.acleddata.com/{data}/{command}	application/json
XML	https://api.acleddata.com/{data}/{command}.xml	text/xml
CSV	https://api.acleddata.com/{data}/{command}.csv	text/csv
TXT	https://api.acleddata.com/{data}/{command}.txt	text/plain

Acled

This data call returns the main dataset

Commands

Read

In order to retrieve the data you must make a GET or POST request to the following URL:

https://api.acleddata.com/acled/read

Returned Data (json only)

Attribute Name	Type	Description
success	boolean	A boolean representation on the success of the call
last_update	int	The number of hours since the last update to the data
count	int	The number of data rows returned
data	array	The rows of data returned. For details of attributes returned in each row, see below.
filename	string	The filename that will be used for csv calls

Returned Data (json, xml, txt, csv)

Attribute Name	Type	Description
data_id	int	A unique id for the row of data
iso	int	A numeric code for each individual country. Referenced here - ISO Country List
event_id_cnty	string	An individual identifier by number and country acronym
event_id_no_cnty	string	An individual numeric identifier
event_date	date	The date the event occurred in the format: yyyy-mm-dd
year	int	The year the event occurred.
time_precision	int	A numeric code indicating the level of certainty of the date coded for the event
event_type	string	The type of conflict event
actor1	string	The named actor involved in the event
assoc_actor_1	string	The named actor allied with or identifying ACTOR1

Attribute Name	Type	Description
inter1	int	A numeric code indicating the type of ACTOR1.
actor2*	string	The named actor involved in the event
assoc_actor_2*	string	The named actor allied with or identifying ACTOR2
inter2*	int	A numeric code indicating the type of ACTOR2.
interaction	int	A numeric code indicating the interaction between types of ACTOR1 and ACTOR2
region	string	The region in which the event took place
country	string	The name of the country the event occurred in
admin1	string	The largest sub-national administrative region in which the event took place
admin2	string	The second largest sub-national administrative region in which the event took place
admin3	string	The third largest sub-national administrative region in which the event took place
location	string	The location in which the event took place
latitude	decimal	The latitude of the location
longitude	decimal	The longitude of the location
geo_precision	int	A numeric code indicating the level of certainty of the location coded for the event
source	string	The source of the event report
source_scale	string	The scale of the source
notes	string	A short description of the event
fatalities	int	The number of reported fatalities which occurred during the event
timestamp	int / date	The unix timestamp this entry was last updated
iso3		A 3 character code representation of each country

* These attributes will be returned as a new data row if export type is monadic.

Query Filters

Each field can be searched to filter the returned data. By default each field will search by = or LIKE based on the table below. This can be changed by sending a new query string name value pair, where the name has '_where' appended to it. The following table shows the default query type that will be used by each field.

Query Name	Type	Query String
data_id	=	?data_id={number}
iso	=	?iso={number}
event_id_cnty	LIKE	?event_id_cnty={text}
event_id_no_cnty	LIKE	?event_id_no_cnty={text}
event_date	=	?event_date={yyyy-mm-dd}
year	=	?year={yyyy}
time_precision	=	?time_precision={number}
event_type	LIKE	?event_type={text}
actor1	LIKE	?actor1={text}
assoc_actor_1	LIKE	?assoc_actor_1={text}
inter1	=	?inter1={number}
actor2	LIKE	?actor2={text}
assoc_actor_2	LIKE	?assoc_actor_2={text}
inter2	=	?inter2={number}
interaction	=	?interaction={number}
region	=	?region={number}
country	LIKE	?country={text}
admin1	LIKE	?admin1={text}
admin2	LIKE	?admin2={text}
admin3	LIKE	?admin3={text}
location	LIKE	?location={text}
latitude	=	?latitude={number}
longitude	=	?longitude={number}
geo_precision	=	?geo_precision={number}

Query Name	Type	Query String
source	LIKE	?source={text}
source_scale	LIKE	?source_scale={text}
notes	LIKE	?notes={text}
fatalities	=	?fatalities={number}
timestamp	>=	?timestamp={number/yyyy-mm-dd}
export_type	=	?export_type={text}
iso3	=	?iso3={text}

References

For some attributes a number is required instead of text. The following reference tables provides the numeric code to be used for certain content.

inter 1 / inter 2	Numeric Code
State Forces	1
Rebel Forces	2
Militia Groups	3
Communal / Identity Groups	4
Rioters	5
Protesters	6
Civilians	7
Foreign / Others	8

region	Numeric Code
Western Africa	1
Middle Africa**	2
Eastern Africa	3
Southern Africa	4
Northern Africa	5
Central Africa	6
Southern Asia	7

region	Numeric Code
Western Asia	8
South-Eastern Asia	9
South Asia	10
Middle East	11

** depreciated - use Central Africa instead

- The ISO country list can be viewed here - ISO Country Link
- All LIKE queries will include a wildcard before and after the entered text.
- Multiple queries can be searched with name/value pairs separated by &. Each field will be searched using AND so all arguments must match for data to be returned.
- More complex queries can be searched to include the OR clause. See Complex Queries below.
- If export_type is not included it will by dyadic. For monadic export you will need to include the export_type query.

To change the default query type you can add an additional name/value pair using the query name and appending '_where' to the query name. The query value could be LIKE or %3D for '='. Additional types of '<', '>' and 'BETWEEN may also be used, representing less than, greater than and between. The between requires the query name value to separate the 2 values with a |.

Example:

?event_id_cnty={text}&event_id_cnty_where=%3D
?event_date={yyyy-mm-dd|yyyy-mm-dd}&event_date_where=BETWEEN

Limit Query & Pagination

By default there is a limit of 500 rows of data returned, 1000 rows if export_type = monadic. You can use the limit query name to change the default number. Setting limit as 0 will return all relevant data. It is likely returning all data will cause a timeout error and we therefore recommend using the page query instead. Each page will return 500 (1000 for monadic) rows of data.

Example:

?limit=100 will return 100 rows of data (200 for monadic).

?page=1 will return the first 500 rows of data (1000 for monadic)

?page=2 will return the next 500 rows of data (1000 for monadic)

Limit Fields Returned

By default all fields will be returned for each line of data. You can use the field query name to change the field items returned. Multiple fields can be requested by separating each one with a pipe (|).

Example:
?field=iso will return just the iso field.
?field=iso|fatalities will return the iso and fatalities data for each row.

Complex Queries

By default all fields must match for the data to be returned. In some instances more complex queries may be required to use the OR clause. This is possible by separating the fields to join, by OR, with :OR: text. The main query item will be the first item in the join, followed by the other items split with :OR: . These can be used with other queries too.

Example:
?field={text}:OR:field2={text2}:OR:field3={text3} will return data where field = text OR field2 = text2 OR field3 = text3.

?field={text}:OR:field2={text2}&country={country} will return data where field = text OR field2 = text2 AND country = country.

Actor

This data call returns the actors

Commands

Read

In order to retrieve the data you must make a GET or POST request to the following URL:

https://api.acleddata.com/actor/read

Returned Data (json only)

Attribute Name	Type	Description
success	boolean	A boolean representation on the success of the call
last_update	int	The number of hours since the last update to the data
count	int	The number of data rows returned
data	array	The rows of data returned. For details of attributes returned in each row, see below.
filename	string	The filename that will be used for csv calls

Returned Data (json, xml, txt, csv)

Attribute Name	Type	Description
actor_name	string	The name of the actor
first_event_date	date	The date the first event for this actor occurred in the format: yyyy-mm-dd
last_event_date	date	The date the last event for this actor occurred in the format: yyyy-mm-dd
event_count	int	The number of events that have occurred with this actor

Query Filters

Each field can be searched to filter the returned data. By default each field will search by = or LIKE based on the table below. This can be changed by sending a new query string name value pair, where the name has '_where' appended to it. The following table shows the default query type that will be used by each field.

Query Name	Type	Query String
actor_name	LIKE	?actor_name={text}
first_event_date	=	?first_event_date={yyyy-mm-dd}
last_event_date	=	?last_event_date={yyyy-mm-dd}
event_count	>=	?event_count={number}

- All LIKE queries will include a wildcard before and after the entered text.
- Multiple queries can be searched with name/value pairs separated by &. Each field will be searched using AND so all arguments must match for data to be returned.
- More complex queries can be searched to include the OR clause. See Complex Queries below.

To change the default query type you can add an additional name/value pair using the query name and appending '_where' to the query name. The query value could be LIKE or %3D for '='. Additional types of '<', '>' and 'BETWEEN may also be used, representing less than, greater than and between. The between requires the query name value to separate the 2 values with a |.

Example:

?actor_name={text}&actor_name_where=%3D

?last_event_date={yyyy-mm-dd|yyyy-mm-dd}&last_event_date_where=BETWEEN

Limit Query & Pagination

By default there is a limit of 500 rows of data returned. You can use the limit query name to change the default number. Setting limit as 0 will return all relevant data. It is likely returning all data will cause a timeout error and we therefore recommend using the page query instead. Each page will return 500 rows of data.

Example:
?limit=100 will return 100 rows of data.

?page=1 will return the first 500 rows of data
?page=2 will return the next 500 rows of data

Complex Queries

By default all fields must match for the data to be returned. In some instances more complex queries may be required to use the OR clause. This is possible by separating the fields to join, by OR, with :OR: text. The main query item will be the first item in the join, followed by the other items split with :OR: . These can be used with other queries too.

Example:
?field={text}:OR:field2={text2}:OR:field3={text3} will return data where field = text OR field2 = text2 OR field3 = text3.

?field={text}:OR:field2={text2}&event_count={number} will return data where field = text OR field2 = text2 AND event_count = number.

Actor Type

This data call returns the actor types

Commands

Read

In order to retrieve the data you must make a GET or POST request to the following URL:

https://api.acleddata.com/actortype/read

Returned Data (json only)

Attribute Name	Type	Description
success	boolean	A boolean representation on the success of the call
last_update	int	The number of hours since the last update to the data
count	int	The number of data rows returned
data	array	The rows of data returned. For details of attributes returned in each row, see below.
filename	string	The filename that will be used for csv calls

Returned Data (json, xml, txt, csv)

Attribute Name	Type	Description
actor_type_id	int	The id of the actor type
actor_type_name	string	The name of the actor type
first_event_date	date	The date the first event for this actor type occurred in the format: yyyy-mm-dd
last_event_date	date	The date the last event for this actor type occurred in the format: yyyy-mm-dd
event_count	int	The number of events that have occurred with this actor type

Query Filters

Each field can be searched to filter the returned data. By default each field will search by = or LIKE based on the table below. This can be changed by sending a new query string name value pair, where the name has '_where' appended to it. The following table shows the default query type that will be used by each field.

Query Name	Type	Query String
actor_type_id	=	?actor_type_id={number}
actor_type_name	LIKE	?actor_name={text}
first_event_date	=	?first_event_date={yyyy-mm-dd}
last_event_date	=	?last_event_date={yyyy-mm-dd}
event_count	>=	?event_count={number}

- All LIKE queries will include a wildcard before and after the entered text.
- Multiple queries can be searched with name/value pairs separated by &. Each field will be searched using AND so all arguments must match for data to be returned.
- More complex queries can be searched to include the OR clause. See Complex Queries below.

To change the default query type you can add an additional name/value pair using the query name and appending '_where' to the query name. The query value could be LIKE or %3D for '='. Additional types of '<', '>' and 'BETWEEN may also be used, representing less than, greater than and between. The between requires the query name value to separate the 2 values with a |.

Example:

?actor_type_name={text}&actor_type_name_where=%3D

?last_event_date={yyyy-mm-dd|yyyy-mm-dd}&last_event_date_where=BETWEEN

Complex Queries

By default all fields must match for the data to be returned. In some instances more complex queries may be required to use the OR clause. This is possible by separating the fields to join, by OR, with :OR: text. The main query item will be the first item in the join, followed by the other items split with :OR: . These can be used with other queries too.

Example:
?field={text}:OR:field2={text2}:OR:field3={text3} will return data where field = text OR field2 = text2 OR field3 = text3.

?field={text}:OR:field2={text2}&event_count={number} will return data where field = text OR field2 = text2 AND event_count = number.

Country

This data call returns the countries

Commands

Read

In order to retrieve the data you must make a GET or POST request to the following URL:

https://api.acleddata.com/country/read

Returned Data (json only)

Attribute Name	Type	Description
success	boolean	A boolean representation on the success of the call
last_update	int	The number of hours since the last update to the data
count	int	The number of data rows returned
data	array	The rows of data returned. For details of attributes returned in each row, see below.
filename	string	The filename that will be used for csv calls

Returned Data (json, xml, txt, csv)

Attribute Name	Type	Description
country	string	The name of the country
iso	int	The iso number of the country
iso3	string	The iso3 representation of the country
first_event_date	date	The date the first event for this actor type occurred in the format: yyyy-mm-dd
last_event_date	date	The date the last event for this actor type occurred in the format: yyyy-mm-dd
event_count	int	The number of events that have occurred with this actor type

Query Filters

Each field can be searched to filter the returned data. By default each field will search by = or LIKE based on the table below. This can be changed by sending a new query string name value pair, where the name has '_where' appended to it. The following table shows the default query type that will be used by each field.

Query Name	Type	Query String
country	LIKE	?country={text}
iso	=	?iso={number}
iso3	=	?iso3={text}
first_event_date	=	?first_event_date={yyyy-mm-dd}
last_event_date	=	?last_event_date={yyyy-mm-dd}
event_count	>=	?event_count={number}

- All LIKE queries will include a wildcard before and after the entered text.
- Multiple queries can be searched with name/value pairs separated by &. Each field will be searched using AND so all arguments must match for data to be returned.
- More complex queries can be searched to include the OR clause. See Complex Queries below.

To change the default query type you can add an additional name/value pair using the query name and appending '_where' to the query name. The query value could be LIKE or %3D for '='. Additional types of '<', '>' and 'BETWEEN may also be used, representing less than, greater than and between. The between requires the query name value to separate the 2 values with a |.

Example:

?country={text}&country_where=%3D

?last_event_date={yyyy-mm-dd|yyyy-mm-dd}&last_event_date_where=BETWEEN

Complex Queries

By default all fields must match for the data to be returned. In some instances more complex queries may be required to use the OR clause. This is possible by separating the fields to join, by OR, with :OR: text. The main query item will be the first item in the join, followed by the other items split with :OR: . These can be used with other queries too.

Example:
?field={text}:OR:field2={text2}:OR:field3={text3} will return data where field = text OR field2 = text2 OR field3 = text3.

?field={text}:OR:field2={text2}&event_count={number} will return data where field = text OR field2 = text2 AND event_count = number.

Region

This data call returns the regions

Commands

Read

In order to retrieve the data you must make a GET or POST request to the following URL:

https://api.acleddata.com/region/read

Returned Data (json only)

Attribute Name	Type	Description
success	boolean	A boolean representation on the success of the call
last_update	int	The number of hours since the last update to the data
count	int	The number of data rows returned
data	array	The rows of data returned. For details of attributes returned in each row, see below.
filename	string	The filename that will be used for csv calls

Returned Data (json, xml, txt, csv)

Attribute Name	Type	Description
region	int	The id of the region
region_name	string	The name of the region
first_event_date	date	The date the first event for this actor type occurred in the format: yyyy-mm-dd
last_event_date	date	The date the last event for this actor type occurred in the format: yyyy-mm-dd
event_count	int	The number of events that have occurred with this actor type

Query Filters

Each field can be searched to filter the returned data. By default each field will search by = or LIKE based on the table below. This can be changed by sending a new query string name value pair, where the name has '_where' appended to it. The following table shows the default query type that will be used by each field.

Query Name	Type	Query String
region	=	?region={number}
region_name	LIKE	?region_name={text}
first_event_date	=	?first_event_date={yyyy-mm-dd}
last_event_date	=	?last_event_date={yyyy-mm-dd}
event_count	>=	?event_count={number}

- All LIKE queries will include a wildcard before and after the entered text.
- Multiple queries can be searched with name/value pairs separated by &. Each field will be searched using AND so all arguments must match for data to be returned.
- More complex queries can be searched to include the OR clause. See Complex Queries below.

To change the default query type you can add an additional name/value pair using the query name and appending '_where' to the query name. The query value could be LIKE or %3D for '='. Additional types of '<', '>' and 'BETWEEN may also be used, representing less than, greater than and between. The between requires the query name value to separate the 2 values with a |.

Example:

?region_name={text}&region_name_where=%3D

?last_event_date={yyyy-mm-dd|yyyy-mm-dd}&last_event_date_where=BETWEEN

Complex Queries

By default all fields must match for the data to be returned. In some instances more complex queries may be required to use the OR clause. This is possible by separating the fields to join, by OR, with :OR: text. The main query item will be the first item in the join, followed by the other items split with :OR: . These can be used with other queries too.

Example:
?field={text}:OR:field2={text2}:OR:field3={text3} will return data where field = text OR field2 = text2 OR field3 = text3.

?field={text}:OR:field2={text2}&event_count={number} will return data where field = text OR field2 = text2 AND event_count = number.