

Newsquawk FIND Feed API v1

Version: 1.0
Base URL: <https://newsquawk.com>

Overview

The Newsquawk FIND Feed API provides real-time access to Newsquawk’s continuously updated stream of headlines and market intelligence.

This feed is the core source for live, **time-sensitive updates** — delivering every breaking development, statement, data release, and market move as it happens.

It is designed for applications that require **instant market context**, from automated alert systems to live news dashboards.

In addition to real-time coverage, the FIND Feed also serves as a **historical record**, allowing clients to **search and filter tagged headline data** to identify patterns, review past events, and analyze market-moving stories with full editorial classification.

Headlines Endpoint

Retrieve the latest Newsquawk headlines with optional search and tag filtering.

GET

```
Shell
https://newsquawk.com/headlines.json?code=YOUR_API_KEY
```

Parameter	Location	Description
code	query	Token provisioned for your account
search[tag_query_matches][tag_ids]	query	Comma-separated tag IDs to include or exclude
search[tag_query_matches][operator]	query	Operator: AND , OR , or NOT (to exclude)
search[q]	query	Free-text search across subject/content
per_page	query	Number of results per page
page	query	Page index (start at 1)

Parameter	Location	Description
<code>since_id</code>	query	Only headlines with <code>id</code> greater than this
<code>updated_since</code>	query	Headlines updated after this ISO-8601 timestamp

Example Requests

Exclude “Research” and “Calendar” (using NOT operator):

Shell

```
curl -g 'https://newsquawk.com/headlines.json?code={token}&search%5Btag_query_matches%5D%5Btag_ids%5D=195607%2C205326&search%5Btag_query_matches%5D%5Boperator%5D=NOT'
```

Search for “ECB” and paginate:

Shell

```
curl -G 'https://newsquawk.com/headlines.json' --data-urlencode 'code={token}'
--data-urlencode 'search[q]=ECB' --data-urlencode 'per_page=100' --data-urlencode 'page=1'
```

Headline Schema

Field	Type	Description
<code>id</code>	integer	Unique identifier
<code>published_at</code>	object	Publish time (<code>unix</code> , <code>text</code> , <code>iso_8601</code>)
<code>updated_at</code>	object	Last update time
<code>updated_sequence</code>	integer	Monotonic counter; use for change detection
<code>subject</code>	string	Headline title
<code>channel</code>	string	Asset class related to the headline
<code>section</code>	string	Section classification
<code>content</code>	HTML	Main headline content (may contain markup)

Field	Type	Description
source	string	Source (e.g., Reuters, Bloomberg, WSJ)
slug	string	URL endpoint for headline
hidden	boolean	Typically <code>false</code>
important	boolean	Market-moving flag
highlighted	boolean	Bold/formatting emphasis
rumour	boolean	Speculative or rumour flag
headline_tags	array	Tags linked to this headline

Example Response

JSON

```
[
  {
    "id": 1318205,
    "updated_sequence": 7669954,
    "published_at": {
      "unix": 1719306120,
      "text": "2024-06-25 10:02:00 +0100",
      "iso_8601": "2024-06-25T09:02:00Z"
    },
    "updated_at": {
      "unix": 1719306277,
      "text": "2024-06-25 10:04:37 +0100",
      "iso_8601": "2024-06-25T09:04:37Z"
    },
    "subject": "ECB's Lagarde: ...",
    "channel": "Central Banks",
    "section": "ECB",
    "content": "<p>Headline content ...</p>",
    "source": "Newsquawk",
    "slug": "/headlines/ecb-lagarde-...",
    "hidden": false,
    "important": true,
    "highlighted": false,
    "rumour": false,
    "headline_tags": [
      { "id": 9991, "tag_id": 1234, "name": "ECB", "weight": 0.9 }
    ]
  }
]
```

Change Detection

- Prefer `updated_sequence` over timestamps to avoid collisions under high frequency updates.
 - Store the latest processed `id` or `updated_sequence` to request deltas using `since_id` or `updated_since`.
-

Errors

Code	Meaning	Typical Cause
401	Unauthorized	Missing or invalid token
403	Forbidden	Insufficient access
429	Too Many Requests	Rate limit exceeded
5xx	Server Error	Temporary failure — retry with exponential backoff

Best Practices

- Filter server-side using tag queries to minimize payloads.
 - Cache responses and deduplicate by `id`.
 - Treat `content` as HTML and sanitize in your frontend.
 - Handle 429 and 5xx errors gracefully with retry/backoff.
 - Monitor `updated_sequence` for incremental sync.
-

Support

For access issues or new client tokens, contact:
support@newsquawk.com