

---

# **nineapi Documentation**

***Release 0.1***

**Andrew Dunai**

**Nov 30, 2018**



<b>1</b>	<b>What works?</b>	<b>3</b>
<b>2</b>	<b>Roadmap</b>	<b>5</b>
2.1	Getting started & running tests . . . . .	5
2.2	Simple client example . . . . .	5
2.3	Client . . . . .	6
2.4	Utils . . . . .	7
	<b>Python Module Index</b>	<b>9</b>



So I'm sick and tired with web apps. I want normal adequate desktop apps for all the things.

Same relates to 9GAG. Everyone says there's no official API for 9GAG, but of course there is - you just need to go deeper.

Here's what I did in 20 minutes:

- Decompiling 9GAG app for Android and digging within its JS sources (that's right - JavaScript, not Java: 9GAG app seems to be written in React Native.) to find the logic for signing requests.
- Dumping HTTP traffic with spoofed SSL certificates to see actual request & response bodies.

So far there seem to be multiple domains used including *api.9gag.com*, *ad.9gag.com*, *notify.9gag.com*, *admin.9gag.com* and *comment-cdn.9gag.com*. I was able to make my own requests and retrieve the data I want.

Any fellow coders willing to contribute to this? We would be the first ones to reverse-engineer an actual 9GAG API. How cool is that?



# CHAPTER 1

---

## What works?

---

- Authorization via 9GAG username and password.
- Fetching posts from hot.
- Works for both Python 2.x & 3.x.





## CHAPTER 2

---

### Roadmap

---

- Fetching posts from different categories
- Paginating & loading more posts
- Reading comments
- Writing comments
- Authorization via Facebook & Google.
- Creating posts

### Getting started & running tests

```
pip install -r requirements.txt

export NINEGAG_USERNAME=your_username
export NINEGAG_PASSWORD=your_password

nosetests --verbose
```

### Simple client example

```
# app.py

from nineapi.client import Client, APIException

client = Client()

try:
    client.log_in('zomgtehlazzers', 'secretnope')
```

```
except APIException as e:
    print('Failed to log in:', e)
else:
    for post in client.get_posts():
        print(post)
```

## Client

**class** `nineapi.client.Client` (*app\_id*='com.ninegag.android.app', *log\_level*=20)  
9GAG API Client.

Create API client instance.

### Parameters

- **app\_id** – Application id (defaults to 'com.ninegag.android.app')
- **log\_level** – Logging level (defaults to `logging.INFO`)

### class Services

Represents sub-API URLs.

**get\_posts** (*group*=1, *type*='hot', *count*=10, *entry\_types*=['animated', 'photo', 'video', 'album'],  
*olderThan*=None, *\*\*kwargs*)

Fetch posts.

### Parameters

- **group** – Posts category (defaults to 1)
- **type** – Posts type (defaults to 'hot')
- **count** – Count of posts.
- **entry\_types** – list of strings
- **olderThan** – Last seen post (for pagination) - *str*, *Post* or *None*

**Returns** list of *Post*

**Raises** *APIException*

### is\_authorized

Authorization status.

**log\_in** (*username*, *password*)

Attempt to log in.

### Parameters

- **username** – User login.
- **password** – User password.

**Returns** *True*

**Return type** *bool*

**Raises** *APIException*

**search\_posts\_by\_tag** (*query*, *offset*=0, *count*=10, *entry\_types*=['animated', 'photo', 'video', 'album'], *sort*='asc', *\*\*kwargs*)

Fetch posts that match specific tag.

**Parameters**

- **query** – Posts tag
- **offset** – Offset to start from.
- **count** – Count of posts.
- **entry\_types** – list of strings
- **sort** – sorting order (“asc” or “desc”) - does not seem to work

**Returns** list of *Post*

**Raises** *APIException*

**class** `nineapi.client.Post` (*client, props*)

Represents single post.

**class Types**

Enum for possible post type values.

**get\_media\_url()**

Returns image URL for Photo posts and .WEBM URL for Animated posts.

**get\_top\_comments()**

Retrieves top comments for this post.

**id**

Post ID.

**props**

Dictionary with post data.

**title**

Post title.

**type**

Post type.

**url**

Post url.

## Utils

`nineapi.utils.get_timestamp()`

Generate timestamp in milliseconds.

`nineapi.utils.md5(string)`

Hash string with MD5.

`nineapi.utils.random_sha1()`

Generate random SHA1 string.

`nineapi.utils.random_uuid()`

Generate random UUID string.

`nineapi.utils.sign_request(timestamp, app_id, device_uuid)`

Generate signature for HTTP request.



### n

`nineapi.client`, 6  
`nineapi.utils`, 7



### C

Client (class in nineapi.client), 6  
Client.Services (class in nineapi.client), 6

### G

get\_media\_url() (nineapi.client.Post method), 7  
get\_posts() (nineapi.client.Client method), 6  
get\_timestamp() (in module nineapi.utils), 7  
get\_top\_comments() (nineapi.client.Post method), 7

### I

id (nineapi.client.Post attribute), 7  
is\_authorized (nineapi.client.Client attribute), 6

### L

log\_in() (nineapi.client.Client method), 6

### M

md5() (in module nineapi.utils), 7

### N

nineapi.client (module), 6  
nineapi.utils (module), 7

### P

Post (class in nineapi.client), 7  
Post.Types (class in nineapi.client), 7  
props (nineapi.client.Post attribute), 7

### R

random\_sha1() (in module nineapi.utils), 7  
random\_uuid() (in module nineapi.utils), 7

### S

search\_posts\_by\_tag() (nineapi.client.Client method), 6  
sign\_request() (in module nineapi.utils), 7

### T

title (nineapi.client.Post attribute), 7

type (nineapi.client.Post attribute), 7

### U

url (nineapi.client.Post attribute), 7