

---

# **docket-python-client Documentation**

*Release 0.0.1*

**Austin Jackson**

**Apr 29, 2019**



---

## Contents

---

<b>1</b>	<b>Docket Python Client</b>	<b>1</b>
1.1	Installation . . . . .	1
1.2	Example . . . . .	1
1.3	Documentation . . . . .	2
<b>2</b>	<b>docketapi package</b>	<b>3</b>
2.1	Module contents . . . . .	3
	<b>Python Module Index</b>	<b>5</b>



---

## Docket Python Client

---

This is a Python client API for [Docket](#) which is a RESTful API for [Stenographer](#).

This library is primarily for use with [ROCK NSM](#) to easily automate retrieval of network traffic for post-process analysis.

### 1.1 Installation

```
pip install docketapi --user
```

### 1.2 Example

```
from docketapi import DocketClient

# create a client
docket = DocketClient('https://rock_nsm_url', 'username', 'password', verify=False)

# perform a query
my_query = docket.query(
    after='2019-04-20T21:07:59.689Z',
    before='2019-04-30T21:07:59.689Z',
    host=['151.101.68.223'],
    proto_name='TCP',
    port=['443']
)
```

(continues on next page)

(continued from previous page)

```
# retrieve pcap
pcap = docket.get_pcap(my_query)

# save pcap
docket.save_pcap(pcap, filename='my_traffic.pcap')
```

## 1.3 Documentation

See the [Docs on RTD](#) for full documentation.

## 2.1 Module contents

docketapi

**class** docketapi.**DocketClient** (*base\_url, username, password, verify=True, proxies=None*)

Bases: object

Docket API Client

Core class used for interacting with the Docket RESTful API.

**Args:** *base\_url* (str): URL pointing to the ROCK NSM (or Docket) instance *username* (str): Authentication username *password* (str): Authentication password *verify* (bool): Verify SSL (ignored on HTTP). Disable to use self-signed certificates *proxies* (dict): Optional requests-style proxies dict

**Attributes:** *base\_url* (str): Full RFC-1738 URL pointing to the ROCK NSM (or Docket) instance *query\_url* (str): URL pointing to Docket query endpoint *username* (str): Authentication username *session* (requests.sessions.Session): Requests session used for all outbound requests

Examples:

```
from docketapi import DocketClient

# create a client
docket = DocketClient('https://rock_nsm_url', 'username', 'password',
    ↪verify=False)

# perform a query
my_query = docket.query(
    after='2019-04-20T21:07:59.689Z',
    before='2019-04-30T21:07:59.689Z',
    host=['151.101.68.223'],
    proto_name='TCP',
    port=['443']
)
```

(continues on next page)

(continued from previous page)

```
# retrieve pcap
pcap = docket.get_pcap(my_query)

# save pcap
docket.save_pcap(pcap, filename='my_traffic.pcap')
```

**get\_pcap** (*query\_result*)

Docket get PCAP, returns raw data

**Args:** *query\_result* (dict): Response from a docket query

**query** (*\*\*kwargs*)

Docket query

**Args:** *after* (str): After datetime in ISO-8601 format (e.g. '2019-04-20T21:07:59.689Z') *before* (str): Before datetime in ISO-8601 format (e.g. '2019-04-20T21:07:59.689Z') *host* (list): List of IP addresses to filter on (e.g. ['192.168.1.1']) *net* (list): List of CIDR notation networks to filter on (e.g. ['192.168.1.0/24']) *port* (list): List of Ports to filter on (e.g. ['22']) *proto\_name* (str): TCP, UDP, or ICMP

**save\_pcap** (*pcap, filename='merged.pcap'*)

Docket save PCAP to disk

**Args:** *pcap* (str): Raw pcap data, response from docket get PCAP *filename* (str): Optional filename to save as, default is merged.pcap



**d**

docketapi, 3



## D

docketapi (*module*), 3

DocketClient (*class in docketapi*), 3

## G

get\_pcap () (*docketapi.DocketClient method*), 4

## Q

query () (*docketapi.DocketClient method*), 4

## S

save\_pcap () (*docketapi.DocketClient method*), 4