

---

# **python-proteus Documentation**

*Release 0.1*

**Stephan Adig**

**Nov 20, 2017**



---

## Contents

---

<b>1</b>	<b>proteus Package</b>	<b>3</b>
<b>2</b>	<b>Examples</b>	<b>13</b>
<b>3</b>	<b>Indices and tables</b>	<b>15</b>
	<b>Python Module Index</b>	<b>17</b>



Contents:



## 1.1 proteus Package

### 1.1.1 proteus.api Package

#### proteus.api.client Module

#### Class ProteusClientApi

```
class proteus.api.client.ProteusClientApi (api_url=None, api_user=None,  
                                           api_password=None)
```

Bases: object

Low Level Proteus SOAP Wrapper Class

Constructor

#### Parameters

- *api\_url* : string
- *api\_user* : string
- *api\_password* : string

#### Example:

```
>>> from proteus.api import ProteusClientApi  
>>> pc=ProteusClientApi(  
    'http://proteus.domain.tld/',  
    'username',  
    'password')
```

#### **\_connect** ()

Establish connection to Proteus SOAP Service

**`_disconnect()`**

Disconnect from Proteus SOAP Service

**`_get_entities(parent_id, entity_type, start=1, count=1)`**

Get a list of Proteus Entities

**Parameters**

- `parent_id` : int
- `entity_type` : string [ use one of the TYPE\_\* constants from proteus.api.constants ]
- `start` : int [1-based]
- `count` : int

**Returns** `APIEntityArray`

**`_get_entity_by_name(parent_id, entity_name, entity_type)`**

Wrapper for Proteus SOAP API Method `getEntityByName`

**Parameters**

- `parent_id` : int
- `entity_name` : string
- `entity_type` : string [ use one of the TYPE\_\* constants from proteus.api.constants ]

**Returns** `APIEntity`

**`is_valid_connection()`**

Checks if the client is connected and authenticated

**`login()`**

Connect and login

**Example:**

```
>>> from proteus.api import ProteusClientApi
>>> pc=ProteusClientApi(
    'http://proteus.domain.tld/',
    'username',
    'password')
>>> pc.login()
```

**`logout()`**

Logout and disconnect

**Example:**

```
>>> from proteus.api import ProteusClientApi
>>> pc=ProteusClientApi(
    'http://proteus.domain.tld/',
    'username',
    'password')
>>> pc.login()
>>> ...
>>> pc.logout()
```



## Class ProteusClient

```
class proteus.api.client.ProteusClient (api_url=None, api_user=None, api_password=None,
                                         config_name=None)
```

Bases: *proteus.api.client.ProteusClientApi*

Usable Proteus Client

### Parameters

- *api\_url* : string
- *api\_user* : string
- *api\_password* : string
- *config\_name* : string

### Example:

```
>>> from proteus.api import ProteusClientApi
>>> pc=ProteusClientApi(
    'http://proteus.domain.tld/',
    'username',
    'password',
    'proteus_configuration_object_name')
```

`_get_configuration()`

`get_configuration()`

`get_dns()`

**Configuration**

Configuration Property

**DNS**

DNS Class Property

## Constants Module

### Proteus Configuration Constant

**TYPE\_CONFIGURATION**

Corresponding Proteus SOAP API Type: *Configuration*

### Proteus DNS Constants

**TYPE\_VIEW**

Corresponding Proteus SOAP API Type: *View*

**TYPE\_ZONE**

Corresponding Proteus SOAP API Type: *Zone*

**TYPE\_HOSTRECORD**

Corresponding Proteus SOAP API Type: *HostRecord*

**TYPE\_MXRECORD**

Corresponding Proteus SOAP API Type: MXRecord

**TYPE\_TXTRECORD**

Corresponding Proteus SOAP API Type: TextRecord

**TYPE\_CNAMERECORD**

Corresponding Proteus SOAP API Type: AliasRecord

**TYPE\_HINFORECORD**

Corresponding Proteus SOAP API Type: HINFORecord

## Proteus IP Constants

**TYPE\_IP4BLOCK**

Corresponding Proteus SOAP API Type: IP4Block

## Special Lists of Proteus Types

**DNS\_ALLTYPES**

Include all Proteus DNS Types

- TYPE\_ZONE
- TYPE\_HOSTRECORD
- TYPE\_MXRECORD
- TYPE\_TXTRECORD
- TYPE\_CNAMERECORD
- TYPE\_HINFORECORD

**ALL\_TYPES**

Include all Proteus Data Types

- TYPE\_CONFIGURATION
- TYPE\_VIEW
- TYPE\_HOSTRECORD
- TYPE\_MXRECORD
- TYPE\_TXTRECORD
- TYPE\_CNAMERECORD
- TYPE\_HINFORECORD
- TYPE\_IP4BLOCK

## Module DNS

### Class DNS

**class** `proteus.api.dns.DNS` (*proteus\_client=None*)

Bases: `object`

Proteus DNS Management Class

#### Parameters

- *proteus\_client* : instance of `proteus.api.client.ProteusClient`

**`_find_zone`** (*zonename, view=None, view\_name=None*)

Find last zone from zonename

#### Parameters

- **zonename** (*str*) – Zonename i.e. ‘subdomain.domain.tld’
- **view** (`proteus.objects.apientity.View`) – View Object (can be None when *view\_name* is not None)
- **view\_name** (*str*) – View Name (can be None, when *view* is not None)

**Returns** `proteus.objects.apientity.Zone`

See:<sup>1</sup>

**`_get_record`** (*hostname, zonename, view=None, view\_name=None, rec\_type='HostRecord'*)

Generic method to retrieve the Proteus Resource Records

#### Parameters

- *hostname* : string
- *zonename* : string
- *view* : `proteus.objects.apientity.View`
- *view\_name* : string
- *rec\_type* : string [ should be one of `proteus.api.constants.DNS_ALLTYPES` ]

#### Returns

- **Depends on the result of Proteus call but can be one of these:**

- `proteus.objects.apientity.HostRecord`
- `proteus.objects.apientity.MXRecord`
- `proteus.objects.apientity.TXTRecord`
- `proteus.objects.apientity.HINFORecord`
- `proteus.objects.apientity.CNAMERecord`
- `proteus.objects.apientity.SRVRecord`

See:<sup>1</sup>

**`_get_records_by_zone`** (*zone=None, record\_type='Zone'*)

Retrieve a list of Resource Records from Proteus

#### Parameters

---

<sup>1</sup> Private Method, only for use inside the class

- **zone** (*proteus.objects.apientity.Zone*) – Zone
- **record\_type** (str (use constants from *proteus.api.constants*)) – Record type to retrieve

**Returns**

- **Depending on the input type it can return:**
  - *proteus.objects.apientity.HostRecord*
  - *proteus.objects.apientity.MXRecord*
  - *proteus.objects.apientity.TXTRecord*
  - *proteus.objects.apientity.SRVRecord*
  - *proteus.objects.apientity.HINFORecord*

**get\_cname\_record** (*hostname, zonename, view=None, view\_name=None*)

Retrieve CNAME Record from Proteus

**Parameters**

- **hostname** (*str*) – the hostname
- **zonename** (*str*) – Name of the Zone i.e. ‘subzone.domain.tld’
- **view** (*proteus.objects.apientity.View*) – View name (can be None when *view\_name* is not None)
- **view\_name** (*str*) – View Name (can be None when *view* is not None)

**Returns** *proteus.objects.apientity.CNAMERecord*

**get\_hinfo\_record** (*hostname, zonename, view=None, view\_name=None*)

Retrieve HINFO Record from Proteus

**Parameters**

- **hostname** (*str*) – the hostname
- **zonename** (*str*) – Name of the Zone i.e. ‘subzone.domain.tld’
- **view** (*proteus.objects.apientity.View*) – View name (can be None when *view\_name* is not None)
- **view\_name** (*str*) – View Name (can be None when *view* is not None)

**Returns** *proteus.objects.apientity.HINFORecord*

**get\_host\_record** (*hostname, zonename, view=None, view\_name=None*)

Retrieve Host Record from Proteus

**Parameters**

- **hostname** (*str*) – the hostname
- **zonename** (*str*) – Name of the Zone i.e. ‘subzone.domain.tld’
- **view** (*proteus.objects.apientity.View*) – View name (can be None when *view\_name* is not None)
- **view\_name** (*str*) – View Name (can be None when *view* is not None)

**Returns** *proteus.objects.apientity.HostRecord*

**get\_mx\_record** (*hostname, zonename, view=None, view\_name=None*)

Retrieve Mailexchanger Record from Proteus

**Parameters**

- **hostname** (*str*) – the hostname
- **zonename** (*str*) – Name of the Zone i.e. ‘subzone.domain.tld’
- **view** (*proteus.objects.apientity.View*) – View name (can be None when view\_name is not None)
- **view\_name** (*str*) – View Name (can be None when view is not None)

**Returns** *proteus.objects.apientity.MXRecord*

**get\_srv\_record** (*hostname, zonename, view=None, view\_name=None*)  
Retrieve SRV Record from Proteus

**Parameters**

- **hostname** (*str*) – the hostname
- **zonename** (*str*) – Name of the Zone i.e. ‘subzone.domain.tld’
- **view** (*proteus.objects.apientity.View*) – View name (can be None when view\_name is not None)
- **view\_name** (*str*) – View Name (can be None when view is not None)

**Returns** *proteus.objects.apientity.SRVRecord*

**get\_txt\_record** (*hostname, zonename, view=None, view\_name=None*)  
Retrieve TXT Record from Proteus

**Parameters**

- **hostname** (*str*) – the hostname
- **zonename** (*str*) – Name of the Zone i.e. ‘subzone.domain.tld’
- **view** (*proteus.objects.apientity.View*) – View name (can be None when view\_name is not None)
- **view\_name** (*str*) – View Name (can be None when view is not None)

**Returns** *proteus.objects.apientity.TXTRecord*

**get\_view** (*view\_name*)  
Get the Proteus View

**Parameters**

- *view\_name* : string

**Returns**

- *proteus.objects.apientity.View*

**get\_views** ()  
Get a list of all Views in Proteus

**Returns**

- list of *proteus.objects.apientity.View*

**get\_zone** (*zone\_name=None, view=None, view\_name=None*)  
Get a Zone Record from Proteus

**Parameters**

- *zone\_name* : string

- *view*: *proteus.objects.apientity.View*
- *view\_name*: string

**Returns**

- *proteus.objects.apientity.Zone*

**get\_zone\_list** (*zonename*, *view=None*, *view\_name=None*, *rec\_type=['Zone', 'HostRecord', 'MXRecord', 'TXTRecord', 'AliasRecord', 'HINFORecord', 'SRVRecord']*)

Retrieves a list of resource records for a special zone from Proteus

**Parameters**

- **zonename** (*str*) – Name of the Zone i.e. 'subzone.domain.tld'
- **view** (*proteus.objects.apientity.View*) – View (can be None when *view\_name* is not None)
- **view\_name** (*str*) – Name of the View (can be None when *view* is not None)
- **rec\_type** (*str* (use one of the constants of *proteus.api.constants* or use *DNS\_ALLTYPES*)) – Type of Record to return

**Returns**

- **Depending on the input type it can return:**
  - *proteus.objects.apientity.HostRecord*
  - *proteus.objects.apientity.MXRecord*
  - *proteus.objects.apientity.TXTRecord*
  - *proteus.objects.apientity.SRVRecord*
  - *proteus.objects.apientity.HINFORecord*
- **or when *rec\_type* is *DNS\_ALLTYPES*:**
  - return a mixed list of all types above

## 1.1.2 objects Package

### objects Package

#### apientity Module

**class** *proteus.objects.apientity.APIObject* (*\*args*, *\*\*kwargs*)

Bases: object

Factory for creating the correct APIEntity Python Objects

**class** *proteus.objects.apientity.CNAMERecord* (*\*args*, *\*\*kwargs*)

Bases: *proteus.objects.apientity.ProteusDataObjects*

**class** *proteus.objects.apientity.Configuration* (*\*args*, *\*\*kwargs*)

Bases: *proteus.objects.apientity.ProteusDataObjects*

**class** *proteus.objects.apientity.DNSOptionsRecord* (*\*args*, *\*\*kwargs*)

Bases: *proteus.objects.apientity.ProteusDataObjects*

**class** *proteus.objects.apientity.HINFORecord* (*\*args*, *\*\*kwargs*)

Bases: *proteus.objects.apientity.ProteusDataObjects*

```
class proteus.objects.apientity.HostRecord(*args, **kwargs)
    Bases: proteus.objects.apientity.ProteusDataObjects

    add(*args, **kwargs)

    delete(*args, **kwargs)

    update(*args, **kwargs)

class proteus.objects.apientity.MXRecord(*args, **kwargs)
    Bases: proteus.objects.apientity.ProteusDataObjects

class proteus.objects.apientity.ProteusDataObjects(*args, **kwargs)
    Bases: object

    add(*args, **kwargs)
        Adding records

        Overwrite this method

    delete(*args, **kwargs)
        Deleting records

        Overwrite this method

    update(*args, **kwargs)
        Updating records

        Overwrite this method

class proteus.objects.apientity.ProteusPropertyObject(properties=None)
    Bases: object

class proteus.objects.apientity.SOARecord(*args, **kwargs)
    Bases: proteus.objects.apientity.ProteusDataObjects

class proteus.objects.apientity.SRVRecord(*args, **kwargs)
    Bases: proteus.objects.apientity.ProteusDataObjects

class proteus.objects.apientity.TXTRecord(*args, **kwargs)
    Bases: proteus.objects.apientity.ProteusDataObjects

class proteus.objects.apientity.View(*args, **kwargs)
    Bases: proteus.objects.apientity.ProteusDataObjects

class proteus.objects.apientity.Zone(*args, **kwargs)
    Bases: proteus.objects.apientity.ProteusDataObjects
```





## CHAPTER 2

---

Examples

---



## CHAPTER 3

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`



**p**

`proteus.api`, 10  
`proteus.objects`, 10  
`proteus.objects.apientity`, 10



## Symbols

`_connect()` (proteus.api.client.ProteusClientApi method), 3

`_disconnect()` (proteus.api.client.ProteusClientApi method), 3

`_find_zone()` (proteus.api.dns.DNS method), 7

`_get_configuration()` (proteus.api.client.ProteusClient method), 5

`_get_entities()` (proteus.api.client.ProteusClientApi method), 4

`_get_entity_by_name()` (proteus.api.client.ProteusClientApi method), 4

`_get_record()` (proteus.api.dns.DNS method), 7

`_get_records_by_zone()` (proteus.api.dns.DNS method), 7

## A

`add()` (proteus.objects.apientity.HostRecord method), 11

`add()` (proteus.objects.apientity.ProteusDataObjects method), 11

`ALL_TYPES` (built-in variable), 6

`APIObject` (class in proteus.objects.apientity), 10

## C

`CNAMERecord` (class in proteus.objects.apientity), 10

`Configuration` (class in proteus.objects.apientity), 10

`Configuration` (proteus.api.client.ProteusClient attribute), 5

## D

`delete()` (proteus.objects.apientity.HostRecord method), 11

`delete()` (proteus.objects.apientity.ProteusDataObjects method), 11

`DNS` (class in proteus.api.dns), 7

`DNS` (proteus.api.client.ProteusClient attribute), 5

`DNS_ALLTYPES` (built-in variable), 6

`DNSOptionsRecord` (class in proteus.objects.apientity), 10

## G

`get_cname_record()` (proteus.api.dns.DNS method), 8

`get_configuration()` (proteus.api.client.ProteusClient method), 5

`get_dns()` (proteus.api.client.ProteusClient method), 5

`get_hinfo_record()` (proteus.api.dns.DNS method), 8

`get_host_record()` (proteus.api.dns.DNS method), 8

`get_mx_record()` (proteus.api.dns.DNS method), 8

`get_srv_record()` (proteus.api.dns.DNS method), 9

`get_txt_record()` (proteus.api.dns.DNS method), 9

`get_view()` (proteus.api.dns.DNS method), 9

`get_views()` (proteus.api.dns.DNS method), 9

`get_zone()` (proteus.api.dns.DNS method), 9

`get_zone_list()` (proteus.api.dns.DNS method), 10

## H

`HINFORecord` (class in proteus.objects.apientity), 10

`HostRecord` (class in proteus.objects.apientity), 10

## I

`is_valid_connection()` (proteus.api.client.ProteusClientApi method), 4

## L

`login()` (proteus.api.client.ProteusClientApi method), 4

`logout()` (proteus.api.client.ProteusClientApi method), 4

## M

`MXRecord` (class in proteus.objects.apientity), 11

## P

`proteus.api` (module), 10

`proteus.objects` (module), 10

`proteus.objects.apientity` (module), 10

`ProteusClient` (class in proteus.api.client), 5

`ProteusClientApi` (class in proteus.api.client), 3

`ProteusDataObjects` (class in proteus.objects.apientity), 11

ProteusPropertyObject (class in proteus.objects.apientity), 11

## S

SOARecord (class in proteus.objects.apientity), 11

SRVRecord (class in proteus.objects.apientity), 11

## T

TXTRecord (class in proteus.objects.apientity), 11

TYPE\_CNAMERECORD (built-in variable), 6

TYPE\_CONFIGURATION (built-in variable), 5

TYPE\_HINFORECORD (built-in variable), 6

TYPE\_HOSTRECORD (built-in variable), 5

TYPE\_IP4BLOCK (built-in variable), 6

TYPE\_MXRECORD (built-in variable), 5

TYPE\_TXTRECORD (built-in variable), 6

TYPE\_VIEW (built-in variable), 5

TYPE\_ZONE (built-in variable), 5

## U

update() (proteus.objects.apientity.HostRecord method), 11

update() (proteus.objects.apientity.ProteusDataObjects method), 11

## V

View (class in proteus.objects.apientity), 11

## Z

Zone (class in proteus.objects.apientity), 11