

# Using Ansible to create droplets on Digital Ocean:

First test your account on Digital Ocean for creating Linux instance manually and then automated the process using Ansible.

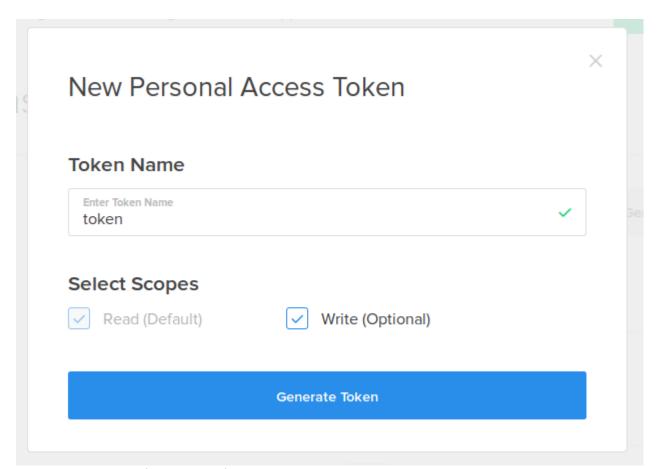
Create an API token on Digital Ocean, using below method.

Next, in the top menu, click on API.

In the Personal Access Tokens section, click the Generate new token button:



You will be taken to the New Personal Access Token screen:



Here, provide the following information:

Your desired token name (for your own reference)



### Select the scope for this token (read or read/write)

#### Then click the Generate Token button.

Your token will be generated and presented to you on your Personal Access Tokens page. The actual token is the long string of numbers and letters, under the name:



Be sure to record your personal access token now. It will not be shown again, for security purposes.

Using the API:

export TOKEN=<token-id>

## Try creating droplets on the Digital Ocean cloud manually using below steps.

### **Example: Create a New Droplet**

To create a new Droplet, send a POST request to /v2/droplets. For a full list of attributes that must be set to successfully create a droplet, see the full documentation. The following example creates an Ubuntu 14.04 droplet called "My-Droplet" in the NYC 2 data center, with 512MB RAM:

```
curl -X POST "https://api.digitalocean.com/v2/droplets" \
    -d'{"name":"My-
Droplet", "region":"nyc2", "size":"512mb", "image":"ubuntu-14-04-x64"}'
    \-H "Authorization: Bearer $TOKEN" \
    -H "Content-Type: application/json"
```

**Note**: This request, like any other request that makes a change to your account, requires that your token has "write" scope assigned to it.

### **Example: Create Multiple Droplets**

You can also create multiple Droplets with the same attributes using a single API request by sending a POST to /v2/droplets. Instead of providing a single name in the request, provide an array of names. The following example creates two Ubuntu 14.04 Droplets, one called "sub-01.example.com" and one called "sub-02.example.com" They both are in the NYC 2 data center, with 512MB RAM:

```
$ curl -X POST "https://api.digitalocean.com/v2/droplets" \
   -d'{"names":["sub-01.example.com","sub-
   02.example.com"],"region":"nyc2","size":"512mb","image":"ubuntu-
   14-04-x64"}' \
   -H "Authorization: Bearer $TOKEN" \
   -H "Content-Type: application/json"
```



### **Using ansible to provision Droplet:**

Configure Ansible,

- 1. sudo apt-get install python-pip
- 2. sudo pip install 'dopy>=0.3.5,<=0.3.5' .... Install 'dopy'
- 3. create a new directory and move the 'ansible.cfg' file.
- 4. Update the .cfg file with below entry.,

```
[defaults]
hostfile = hosts
```

update the hosts file with below entries, here the group name is

```
[digitalocean]
localhost ansible connection=local
```

There are three ways we can tell Ansible about the API token:

- 1. Provide it directly on each DigitalOcean task, using the api\_token parameter.
- 2. Define it as a variable in the playbook or hosts file, and use that variable for the api token parameter.
- 3. Export it as an environment variable, as either DO\_API\_TOKEN or DO API KEY.

Now define the Digital\_Ocean resources in the playbook as written below.

```
digitalocean.yml
---
- hosts: digitalocean

vars:
    do_token:
186a6fd279015640995dce8938ed632020b2c9e6513af57c9cd7473217e280c1
    droplets:
    - droplet-one
    - droplet-two

tasks:
- name: ensure ssh key exists
    user: >
```



```
name={{ ansible user id }}
    generate ssh key=yes
    ssh key file=.ssh/id rsa
- name: ensure key exists at DigitalOcean
 digital ocean: >
    state=present
    command=ssh
   name=my_ssh_key
    ssh pub key={{ lookup('file', '~/.ssh/id rsa.pub') }}
    api token={{ do token }}
  register: my_ssh_key
- name: ensure droplets exist
 digital ocean: >
    state=absent
    command=droplet
   name={{ item }}
    unique name=yes
    size id=512mb
    region id=blr1
    image_id=ubuntu-14-04-x64
    ssh_key_ids={{ my_ssh_key.ssh_key.id }}
    api_token={{ do token }}
  with_items: "{{ droplets }}"
  register: droplet details
```

Run below command to create Droplet on Digital Ocean using Ansible.

```
$ ansible-playbook digitalocean.yml
```