Digital Ocean Setup Overview

The purpose of this document is to set up a Droplet on Digital Ocean. The Droplet will host a website and make it available to the general public.
Successful completion will involve creating several components and have them interact with each other.
There will be:

- A Repository on Github.
- A Droplet on Digital Ocean.
- A sub-domain.

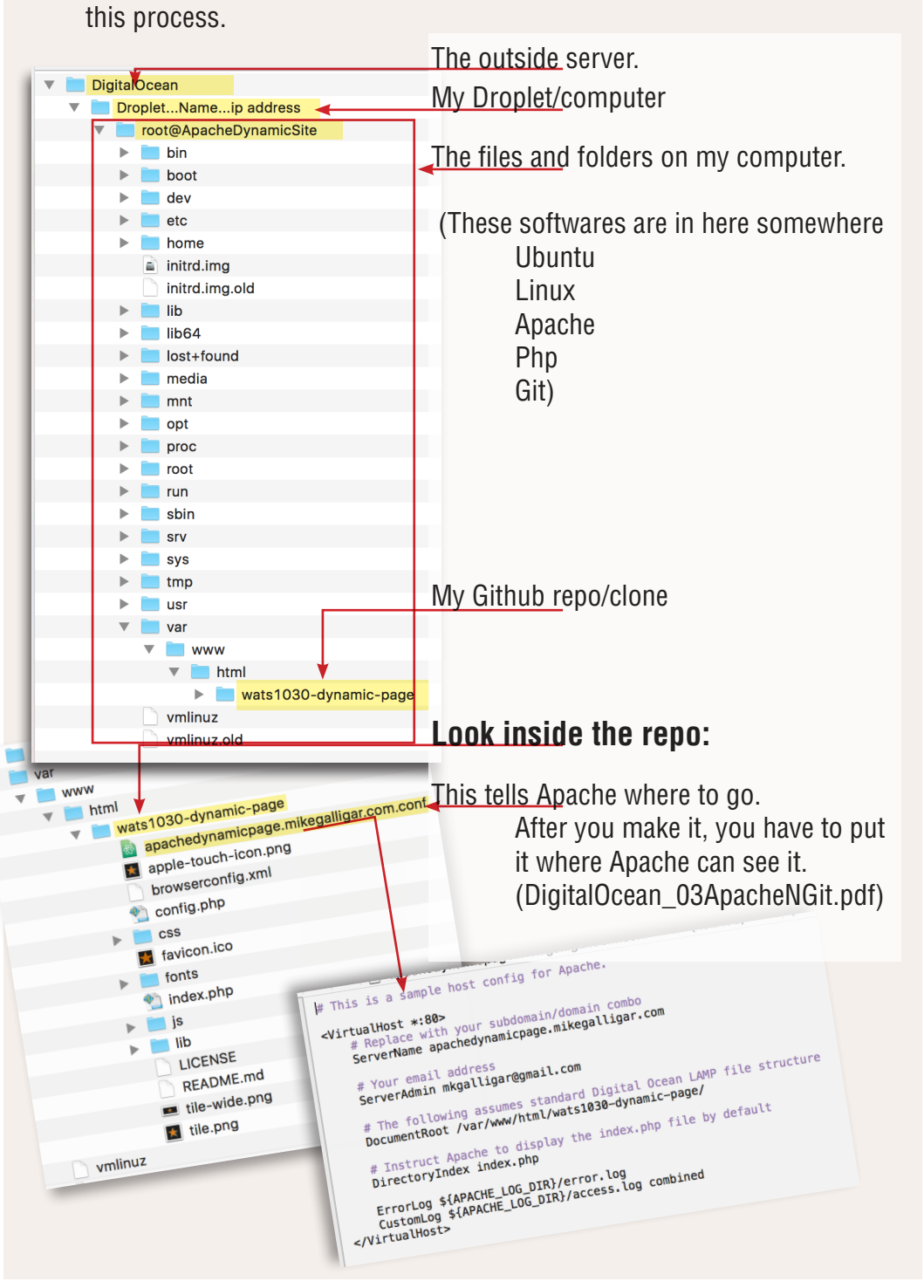
The repository on Github contains the files that make-up a website.
The Droplet will host the website.
The Sub-domain receives a copy of the Droplet's ip address, this will point the Internet to the droplet so the website can be viewed.

To make everything work, the Droplet will need several programs. Some of these programs will be configured to perform the task of presenting the site. Then the website files will be cloned from Github to the Droplet. At this point, a site visitor can enter our sub-domain.domain.com in a browser, and they will be taken to the site. A lot of this configuring will take place in Terminal. We use Terminal to send codes and files to the Droplet so it performs in the way we want.

The Goal:

Create a computer
hosted on an outside server to show our website

This shows the directory structure that is created by going through this process.



The outside server.
My Droplet/computer
The files and folders on my computer.

(These softwares are in here somewhere
Ubuntu
Linux
Apache
Php
Git)

My Github repo/clone

Look inside the repo:

This tells Apache where to go.
After you make it, you have to put it where Apache can see it.
(DigitalOcean_03ApacheNGit.pdf)

The Componets:

Digital Ocean
Runs/Has:
Droplet
SSH Key
ip address
Ubuntu
LAMP
Git
Repo from Github

Github
Runs/Has:
Repository
Copy of Droplet's SSH Key
This allows Git & This Droplet to interact.

Domain
Runs/Has:
Sub-domain
Copy of Droplet's ip address
Because of this, when you type:
sub-domain.domain
You will be taken to see what's in the Droplet/ What the Droplet has been configured to show you.

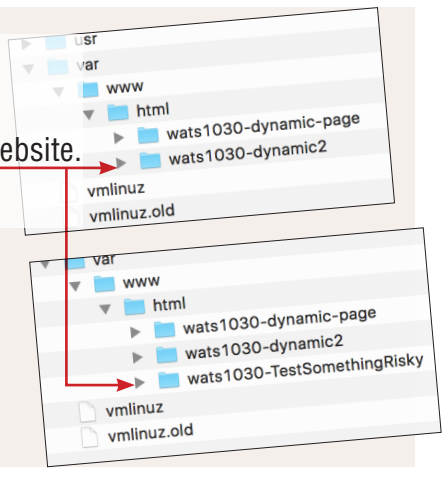
Terminal:

What to do in Terminal?
You have a Droplet, a Sub-domain, and a Github repository. Now what?

If you started correctly the droplet already has Ubuntu and LAMP stack.

1) Install Git on the Droplet.
2) Configure Git.
3) Create SSH Key on the Droplet.
   Copy the public SSH Key to your Github account. This allows the Droplet & Git to interact.
4) Clone the Git repository to the Droplet.
   Change the directory into...~/var/www/html/**Your Clone**
5) Configure the virtual host files,
   so apache shows this repo when:
   sub-domain.domain is typed into a web browser

Add more to your Droplets.
Another repo/sub-domain/website.



If you want to put more repositories in this droplet,
Make a new sub-domain.
With the Droplet's ip address.
Make a clone of the new repository.
Configure Apache using the file:
sub-domain.domain.com.conf

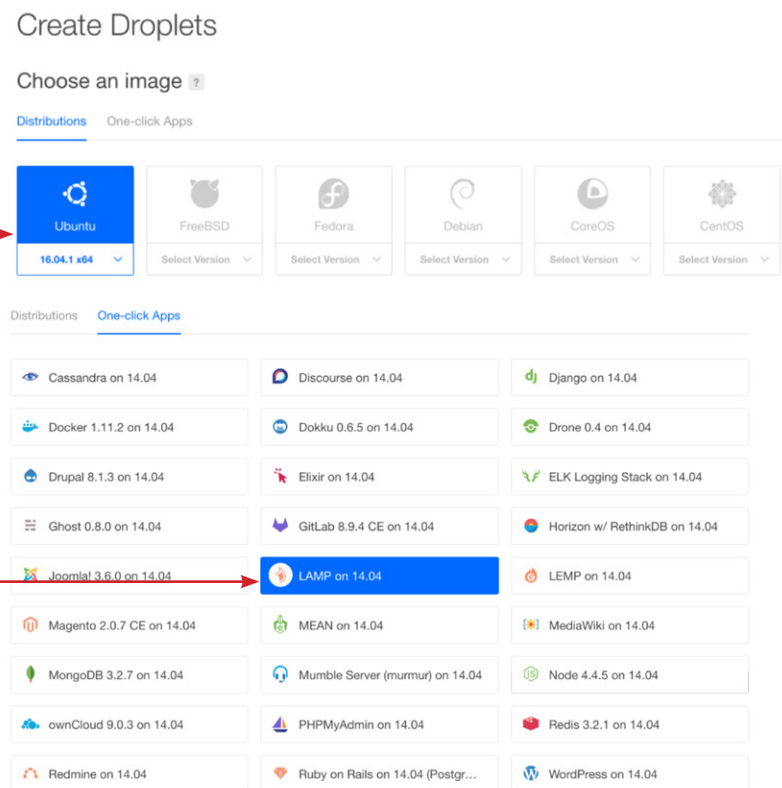
Digital Ocean Create a New Droplet

This sheet is to show how to create, a new Droplet, a new sub-domain, and point the sub-domain to the droplet.

Log into Digital Ocean: https://cloud.digitalocean.com/droplets
This droplet is for a dynamic page. It will need an operating system and a server.

Choose an operating system:

Ubuntu

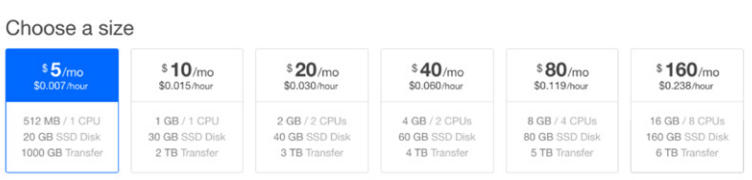


The droplet will need software. Digital Ocean provides a number of options. For now, we'll choose LAMP:

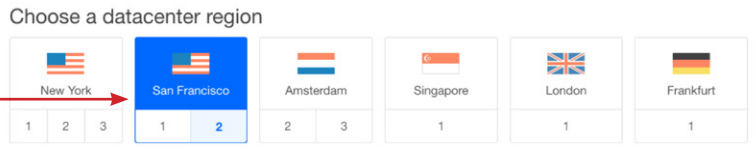
LAMP stands for:

LINUX Operating System
APACHE Web Server
MySQL Database
PHP
http://en.wikipedia.org/wiki/LAMP_(software_bundle)

Choose a size:



Choose a region:

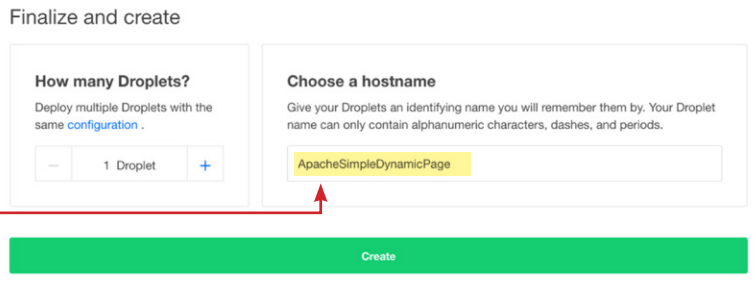


Add an SSH key:

You can make an SSH key for your computer and use that.

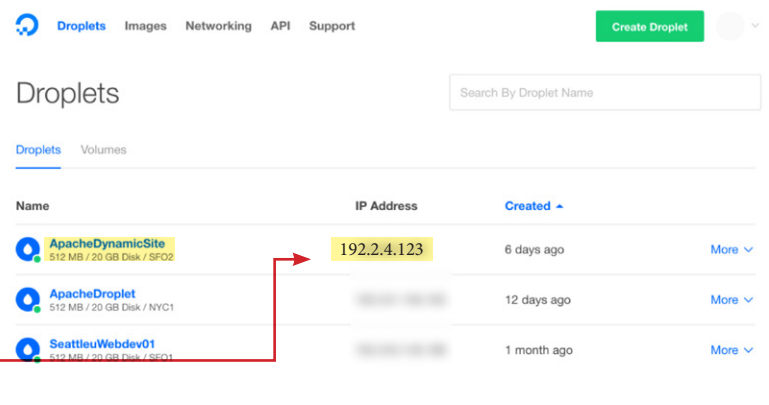
Choose a host name:

Make the host name match the sub-domain. That way it will be easier to remember which things go together.



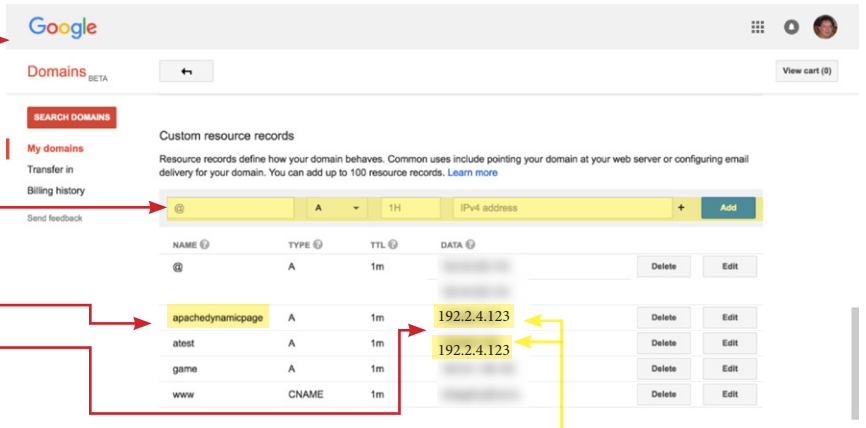
Make note of the ip address of the Droplet.

This will be pasted into the new sub-domain.



Create a sub-domain:

The original domain is made on Google.
Go to: https://domains.google.com



Create a new sub-domain by filling out this row.

Make the sub-domain name match the Droplet name.
Copy the Droplet ip address & paste it here.

After you have a Droplet, you can:

Create multiple directories/websites.
Just point the new sub-domains to the same (Droplet) ip address.

Next, you need to configure Apache inside the new directory in the Droplet

Digital Ocean Configure Git & Apache in Terminal

This sheet is to show how to configure Apache and Git for a new droplet.

Use a command line to connect to the droplet. The following commands will install & configure Git on the droplet. Git needs to be installed at the root level of the Droplet. When done, you can start cloning Git repositories to the droplet.

Note:

Git repos should cloned into ~/var/www/html/**Name-of-repo**
In this example it will be:
root@**NameofDroplet**:/var/www/html/**NameofRepo**

The root in this example uses the name of the Droplet.
root@ApacheDynamicSite:/var/www/html/wats1030-dynamic-page

From Terminal:

Connect to the droplet:
ssh root@**ip address**

INSTALL GIT TO THE DROPLET.
root@ApacheDynamicSite:~# sudo apt-get install git

SET GIT ON THIS DROPLET TO USE THIS NAME.
root@ApacheDynamicSite:~# git config --global user.name **"Name"**

SET GIT ON THIS DROPLET TO USE THIS EMAIL.
root@ApacheDynamicSite:~# git config --global user.email **"email"**

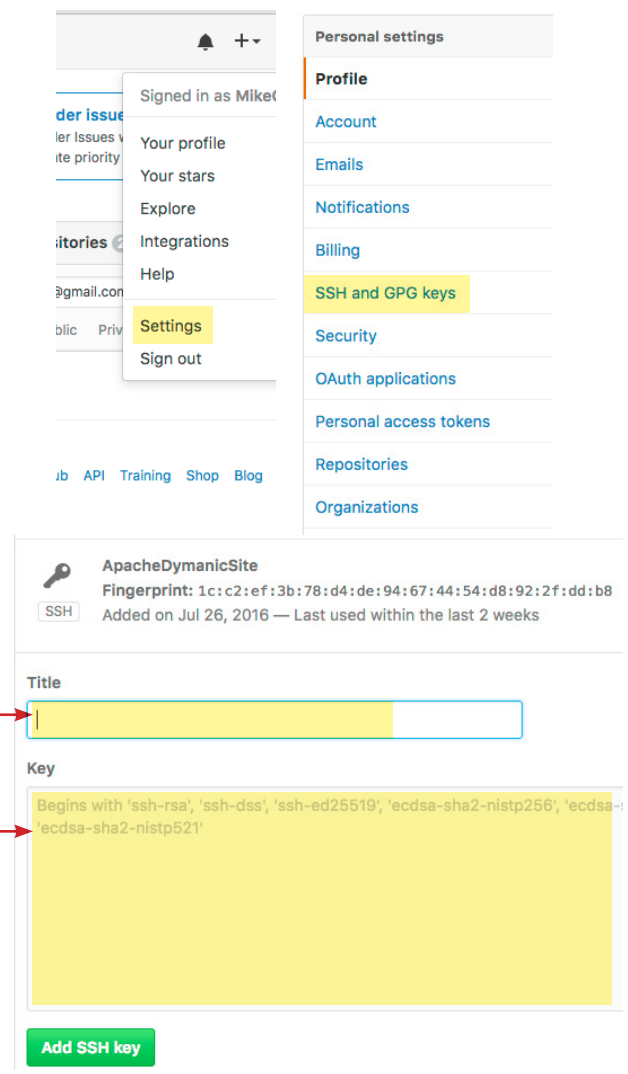
CREATE SSH KEY
root@ApacheDynamicSite:~# ssh-keygen -t rsa -b 4096 -C **"email"**

CREATE SPECIFIC ID FOR SSH
root@ApacheDynamicSite:~# eval "$(ssh-agent -s)"

ADD ID TO THE SSH KEY
root@ApacheDynamicSite:~# ssh-add ~/.ssh/id_rsa

SHOW ME THE PUBLIC SSH KEY. SO I CAN PASTE IT INTO GITHUB.COM
root@ApacheDynamicSite:~# cat ~/.ssh/id_rsa.pub

```
root@ApacheDynamicSite:~# cat ~/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAACAQDBVGMV+yk3CeUv/qR+CFeX0yuodA+6Hz1OzUxfhGUOrO
0AqJOZipsJbWi87dtrB76xW1h+KiPWaM5hy9dMYE1hiYtOnTpIWEJ/PaMrOt241OBwO4NTdnPociNcRfgS
Dho9upBPEx596s2nkRQKhjhXyLa8ykBIEnBllE8I1e+N8TGzgGxAlHlEfdAZQN61u5bWNHfL8AqEijJN2g
EFh4gY1OjlLxy7/vbJ2naZ02O2WMKhIACaK+mlOHm5kMii5kxrcNSvTUEdmYXmx42ptxlnFFAwarpqtOh9
fd7Fd9OeZb0d+5vBkdH+4iM9Jz7NK2X/nku1sF95mLBAzJAfUk9z3NFw/3/cPWemQqeq8CT+cHIDtKa08G
YX6oPd2DPsHq0Yf0t397WAuO7VQzbHikNVs4a2V+YLo84XxlCRGStTnaYE/zfWUUFCj866wi3Jwb5d8abS
9qCv88K66Lu3ZcnP+CM0s722o4iOQEaHjAkRpMOMJY6uNyqrGtRGdg6VEcEvLwnCe1c2jm1KVq5QMGeq8v
HJCx3HOz/qqhqgu/yQl+cdTqG2/VSL9pZeQfQLMJYD1hcMutKBbBdAlq/0Wi2Nt4fasFx5z1mV4r/jo00k
DcMcyfWUCAsACKgg8d4EypxzknjXYbMvL7ml7a2/NQ== mkgalligar@gmail.com
root@ApacheDynamicSite:~#
```



CHANGE TO WEB ROOT DIRECTORY
root@ApacheDynamicSite:~# cd ..
root@ApacheDynamicSite:~# cd /var/www/html

CLONE MY GIT FORK INTO MY DROPLET.
root@**ApacheDynamicSite**:/var/www/html/**wats1030-dynamic-page**#
git clone git@github.com:**GithubUserName**/**GithubRepo**

CHECK YOUR WORK.
root@ApacheDynamicSite:/var/www/html/wats1030-dynamic-page# ls
THE CLONE IS HERE!
wats1030-dynamic-page

Next, Configure Apache to present the info in this droplet/this directory

This exercise contains sample files. Some of these files are the website we want to present, others are used to tell Apache what to do.

Re-name this to this

COPY MOVE & RE-NAME THE DEFAULT APACHE CONFIG FILE
cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-available/**sub-domain.domain.com**.conf
This will put the config file where Apache can find it.

OPEN OUR NEWLY CREATED CONFIG FILE AND MODIFY IT.
Change directory to: ~/etc/apache2/sites-available
root@ApacheDynamicSite:/etc/apache2/sites-available# less **sub-domain.domain**.conf

Tell Apache your **sub-domain.domain**.
Show Apache the path to your files.
Tell Apache which file to display.

```
# This is a sample host config for Apache.

<VirtualHost *:80>
    # Replace with your subdomain/domain combo
    ServerName apachedynamicpage.mikegalligar.com

    # Your email address
    ServerAdmin mkgalligar@gmail.com

    # The following assumes standard Digital Ocean LAMP file structure
    DocumentRoot /var/www/html/wats1030-dynamic-page/

    # Instruct Apache to display the index.php file by default
    DirectoryIndex index.php

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

TELL APACHE, THIS IS THE DEFAULT CONFIG TO USE.
root@ApacheDynamicSite:~# sudo a2ensite **sub-domain.domain.com**.conf

RE-START APACHE
root@ApacheDynamicSite:~# service apache2 reload