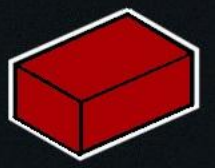
Building a Web Server

For fun, profit and [REDACTED]



Redbrick

DCU's Networking Society

A word on conduct...

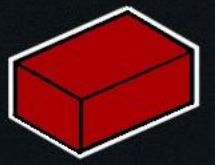
DON'T:

- Touch anyone else's shell. Seriously.
- Host anything illegal, offensive or otherwise immoral on your shiny new server.
- Intentionally disrupt or interrupt the talk. Raise a hand or wait for a pause to ask questions!

DO:

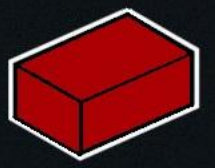
- Bring friends, roommates, pet rocks… All are welcome!
- Ask questions! Don't be shy, we're here to help you!
- Work together! Two heads are (often) better than one.
- Be respectful to us and those around you, everyone is here to have fun.

In short: *Be excellent to each other.*



Redbrick
DCU's Networking Society

If you are not already booted into linux you should do that now

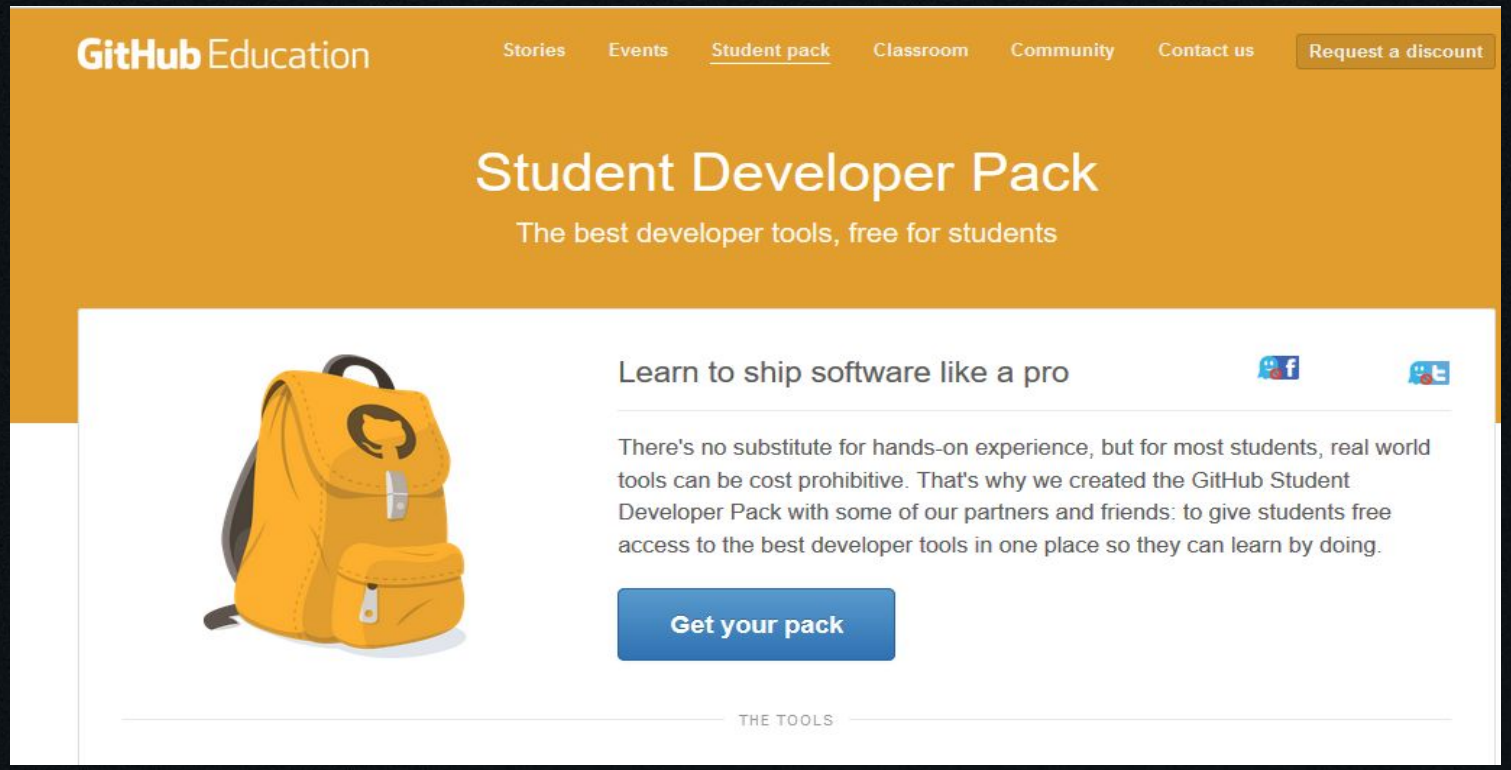


Redbrick
DCU's Networking Society

GitHub Developer Pack

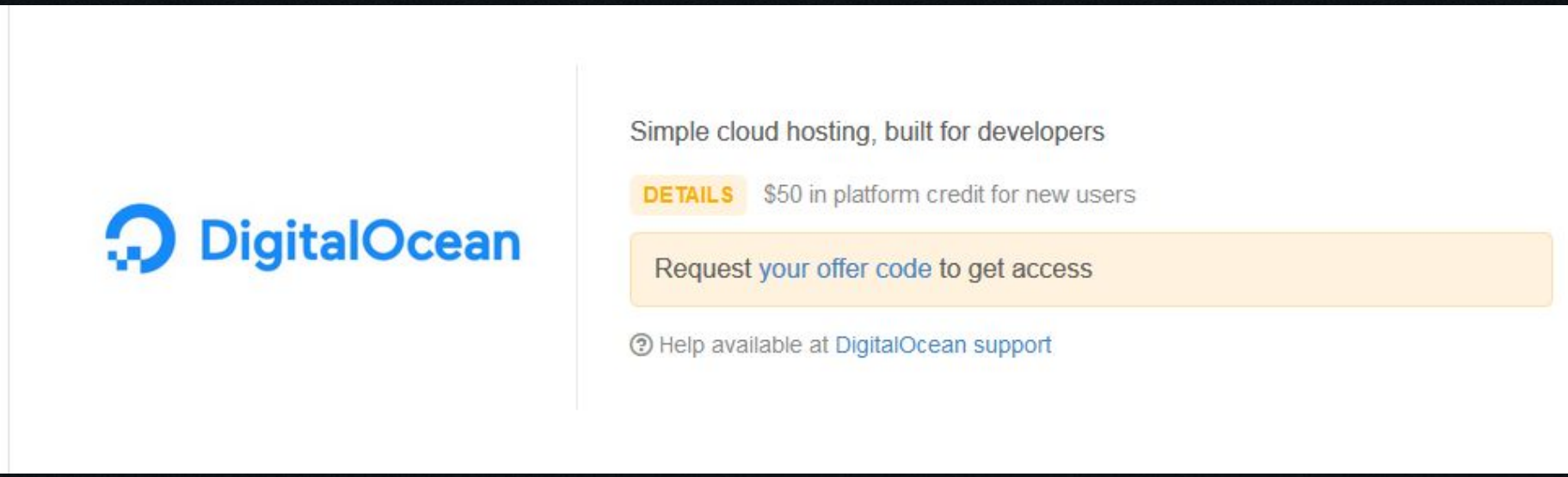
Redbrick
DCU's Networking Society

Did you git yours?



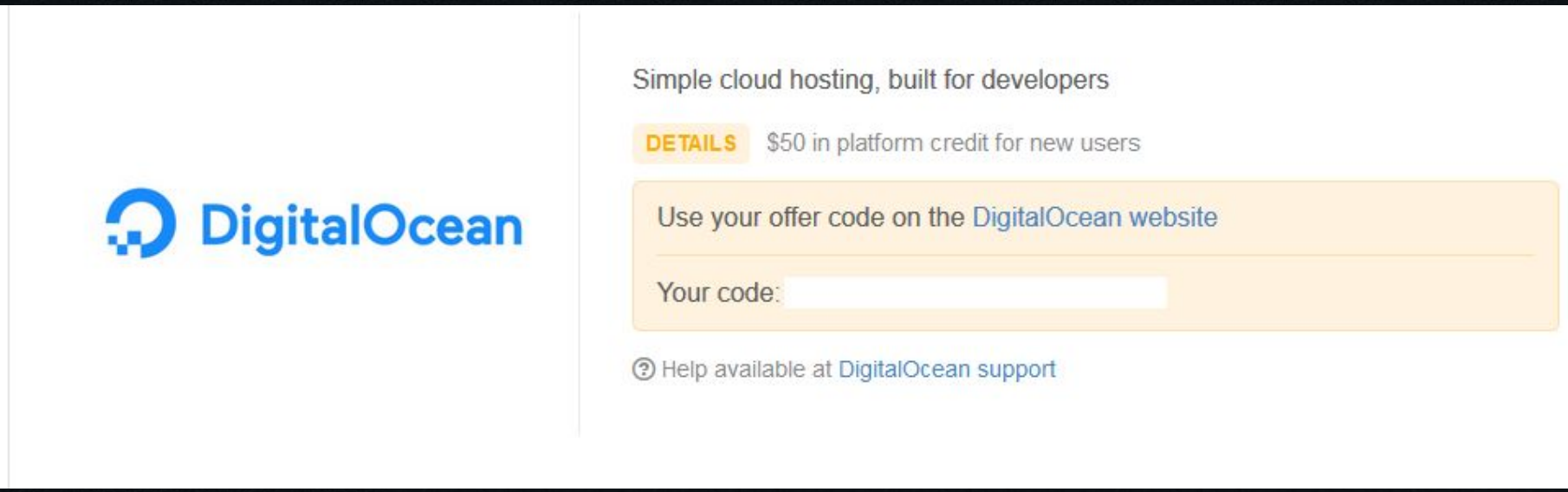
Redbrick
DCU's Networking Society

Find dis one! Click request



Redbrick
DCU's Networking Society

Copy code for later. Click on the website



Redbrick
DCU's Networking Society

Follow the steps

Create an account and enter the promo code when asked for card details.

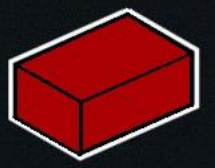
Redbrick
DCU's Networking Society

Ssh keys

So before we go any further lets create an ssh key.
P.s If you have one already this step can be skipped

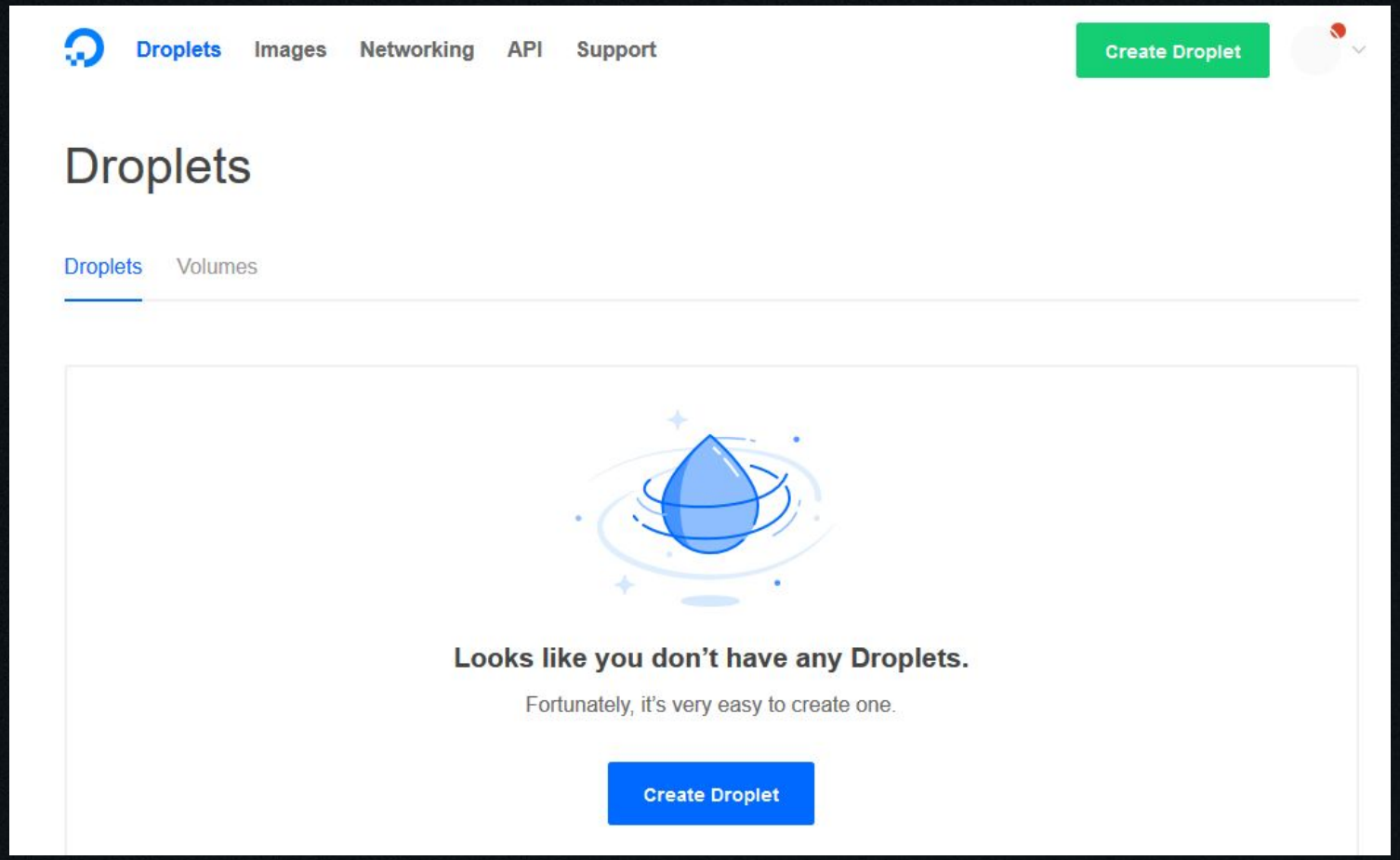
```
ssh-keygen -t rsa -C "your_email@example.com"
```

- You'll be prompted to enter a passphrase if you'd like
- Two files are generated: 1) {name*you*chose} and 2) {name*you*chose}.pub
- We will need these later



Redbrick
DCU's Networking Society

Now lets make a server

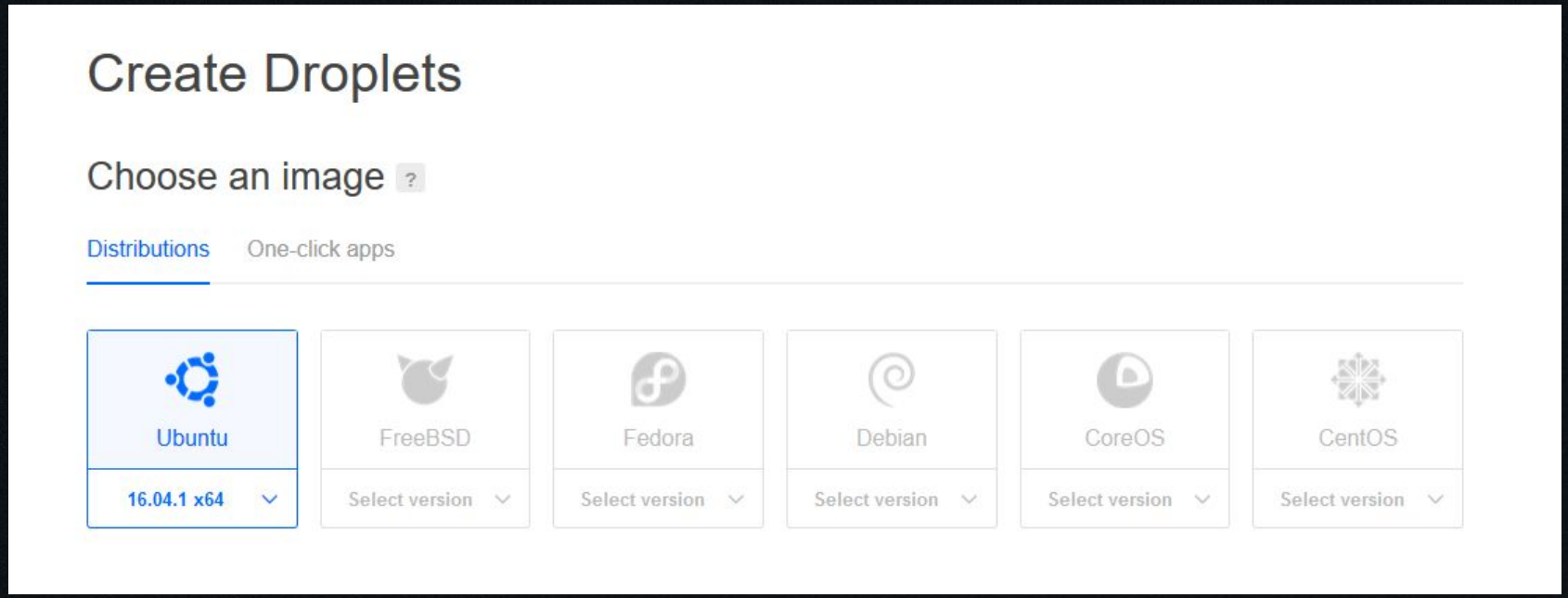


Redbrick
DCU's Networking Society

Choose an operating system

Today we are going to go with Ubuntu because we know it well

There are others operating systems available feel free to experiment later :D

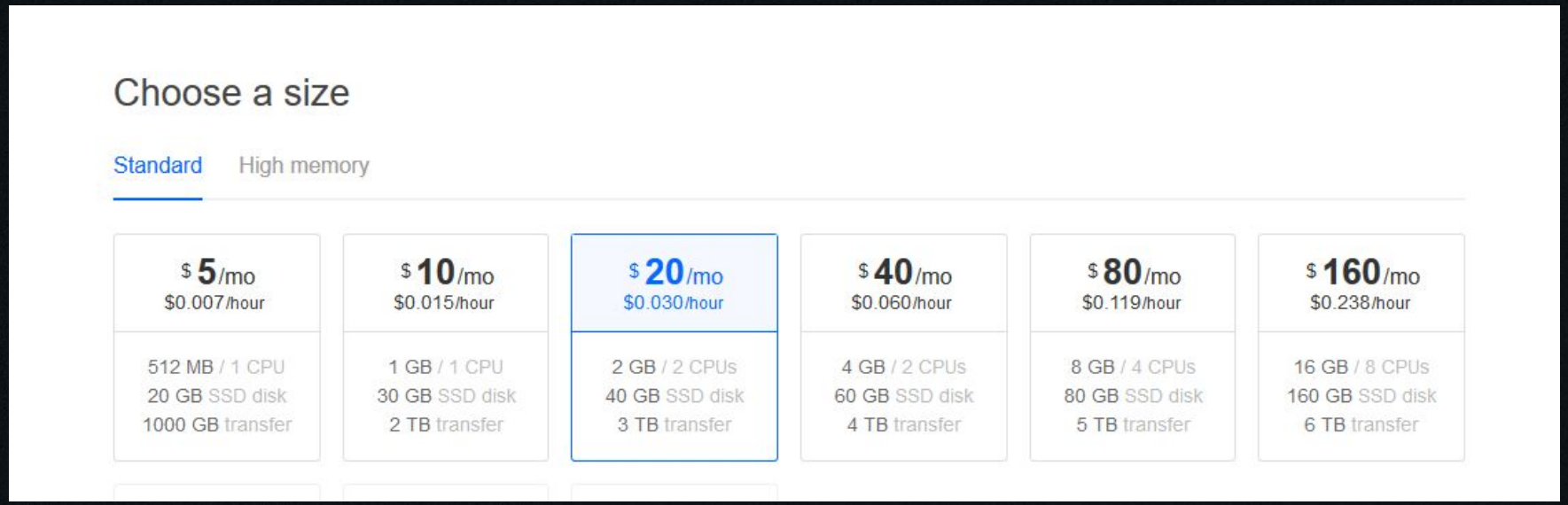


Redbrick
DCU's Networking Society

Big and shiny!

Chose any setting you like. Beware: the more money you spend per month the less time that free $50 will last you.

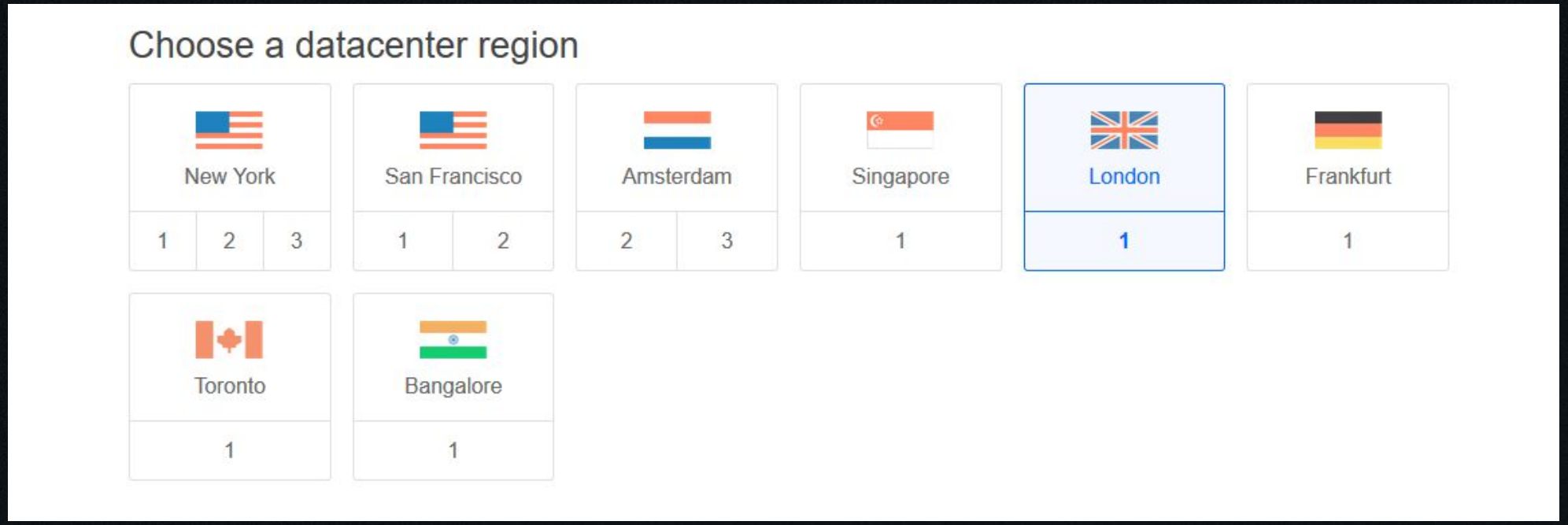
We opted for the $5/month, we don't need anything huge.



Redbrick
DCU's Networking Society

Give your server a home

The Further away it is the longer it will take to connect!



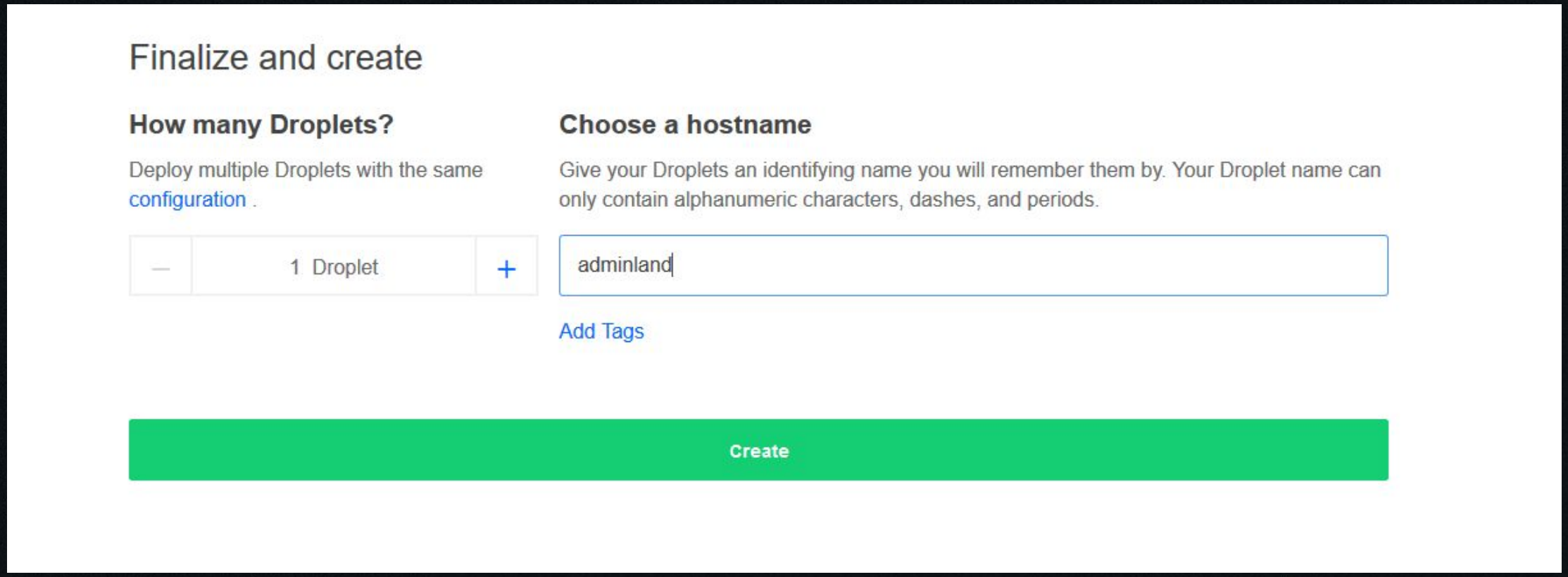
Redbrick
DCU's Networking Society

Remember those keys we made earlier

- Click add SSH keys
- Copy the contents of the .pub part of your SSH keys
- Click save

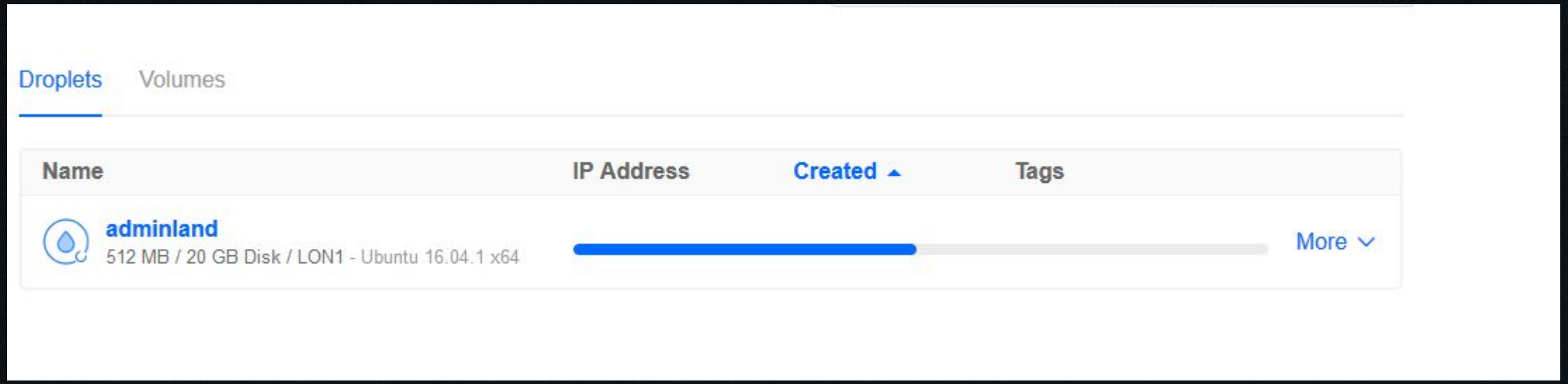
Redbrick
DCU's Networking Society

Naming things is hard



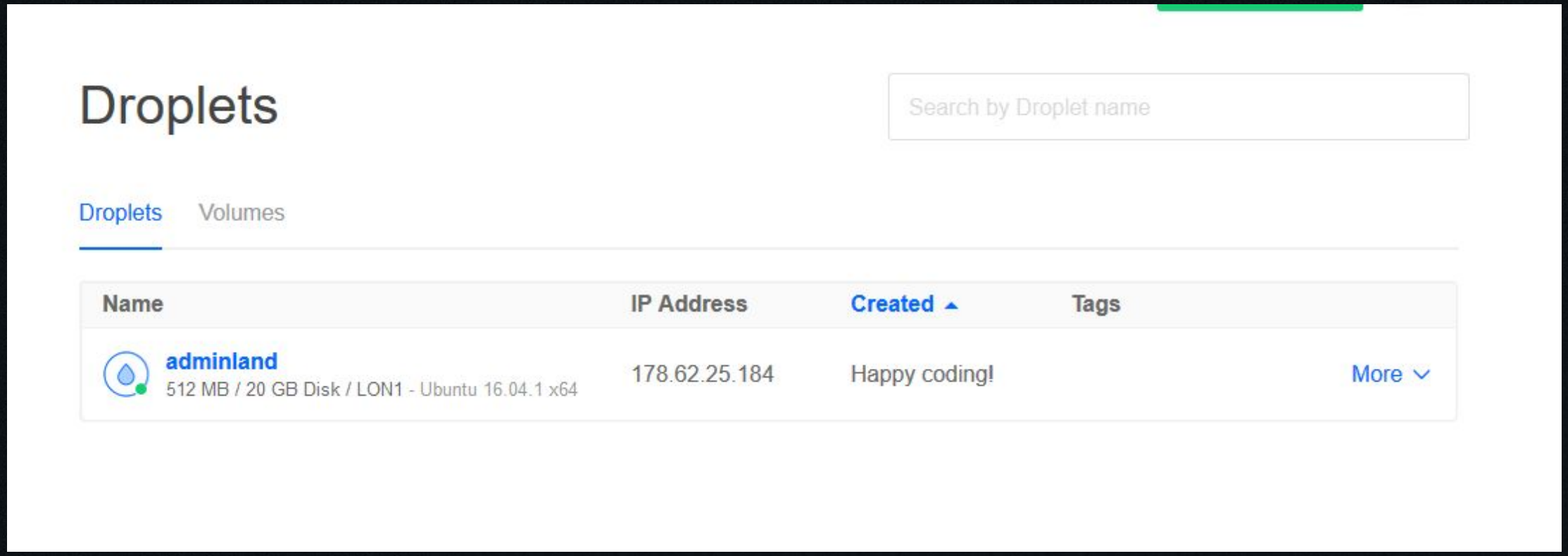
Redbrick
DCU's Networking Society

And now we wait!



Redbrick
DCU's Networking Society

Well that was quick!



Redbrick
DCU's Networking Society

Is anyone totally lost?

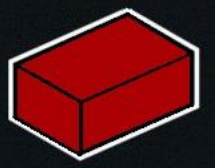
Redbrick
DCU's Networking Society

Now, onto the fun stuff!

First, let's login:

Replace 1.2.3.4 with your ip address from digital ocean.

```
ssh -i ~/.ssh/our_new_key root@1.2.3.4
```

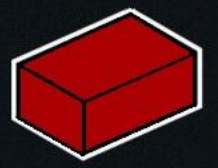


Redbrick

DCU's Networking Society

I don't like this root name I want my own

```
# Add new user "zergless"
sudo adduser zergless

# Copy our installed SSH key to the new user so we can login as them
sudo cp -R .ssh/ /home/zergless/

# Change ownership of evan's SSH directory so he actually has permission to access it
sudo chown -R zergless:zergless /home/zergless/.ssh

# Add zergless to the sudoers (admin) group so he can execute important commands later
sudo usermod -a -G sudo zergless

# Disable root login via password
sudo passwd -l root

# Disable root login via SSH
sudo rm -rf /root/.ssh
```



Redbrick
DCU's Networking Society

Now let’s start installing things!

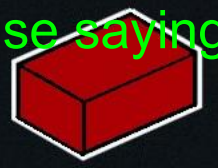
Update our package listing (apt-get is a package manager for ubuntu/debian systems)
```
sudo apt update
```

Install all necessary packages for a LAMP server
```
sudo apt install lamp-server^ libapache2-mod-python
```

You'll be asked to choose a MySQL password

Make sure MySQL is installed securely
```
mysql_secure_installation
```

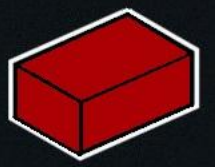
#For the password plugin if you plan on using this server I would advise saying yes and then answer yes to everything else.



Redbrick
DCU’s Networking Society

This was in the space of **1 second**.

```
Feb 13 06:38:17 paphos sshd[10739]: Failed password for root from 116.31.116.34 port 62458 ssh2
Feb 13 06:38:17 paphos sshd[10733]: Failed password for root from 116.31.116.34 port 48839 ssh2
Feb 13 06:38:17 paphos sshd[10745]: Failed password for root from 116.31.116.34 port 54107 ssh2
Feb 13 06:38:17 paphos sshd[10743]: Failed password for root from 116.31.116.34 port 18585 ssh2
Feb 13 06:38:17 paphos sshd[10734]: Failed password for root from 116.31.116.34 port 22850 ssh2
```

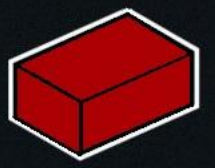


Redbrick
DCU's Networking Society

But does it *do* anything?

Let’s process some requests…

...with python!

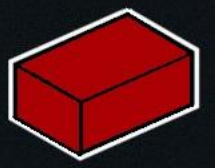


Redbrick

DCU’s Networking Society

Python 3

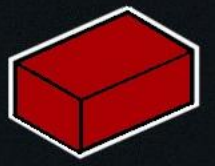
```
# If python 2.7 is installed, delete it
sudo rm /usr/bin/python

# Make python3 the default for when we type the "python"
# command
sudo ln -s /usr/bin/python3 /usr/bin/python
```



Redbrick
DCU's Networking Society

Pip - A python dependency manager

```
# Install pip
sudo apt-get install python3-pip

# Use pip to install the MySQL interface for python
sudo pip3 install pymysql
```



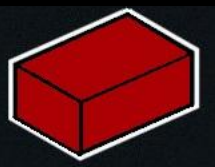
Redbrick
DCU's Networking Society

Put yo scripts in places!

```
# Make a directory to hold all our python scripts
sudo mkdir /var/www/python_scripts

# Disable bad Python multithreading and enable CGI
sudo a2dismod mpm_event
sudo a2enmod mpm_prefork cgi

# Copy the default config and make a backup
sudo cp /etc/apache2/sites-available/000-default.conf
/etc/apache2/sites-available/000-default.conf.backup
```

Redbrick
DCU's Networking Society

```
<VirtualHost *:80>
        # Our files will be located in /var/www/python_scripts
        DocumentRoot /var/www/python_scripts

        # Our server admin's email is zergless@redbrick.dcu.ie
        ServerAdmin zergless@redbrick.dcu.ie

        # The following directives ONLY APPLY to the /var/www/python_scripts directory
        <Directory /var/www/python_scripts>
                # Enable Common Gateway Interface execution
                Options +ExecCGI

                # If "index.py" is present, that's taken as the primary file to load first
                DirectoryIndex index.py

                # Allow everyone to access the scripts in this folder
                Require all granted

                AddHandler mod_python .py
                PythonHandler mod_python.publisher
                PythonDebug On
        </Directory>


        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined

</VirtualHost>
```

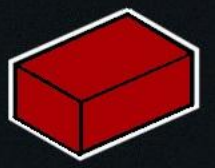


Redbrick

DCU's Networking Society

Now let’s enable our site!

```
# Reload apache's configuration
sudo systemctl apache2 reload
```



Redbrick
DCU’s Networking Society

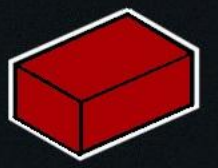
...maybe we should check if it works.

#Create a file ‘index.py’ to test our server

```
sudo vim /var/www/python/index.py
```

#A simple test request:

```
1 def index(req);
2     return "Hello, World! I am a python script!";
```



Redbrick

DCU’s Networking Society

So let’s recap

We made some SSH keys

We installed our server

We added our own user

We installed a lamp stack and hardened our mysql

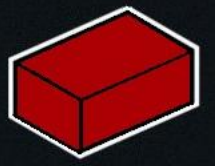
Installed python 3

Installd pip

Made a directory for our python scripts

Allowed apache to execute our scripts

And tested it and if all goes well that should work



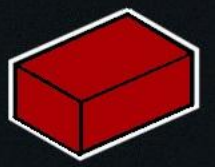
Redbrick

DCU’s Networking Society

Redbrick
DCU's Networking Society

Well not quite

- We want to secure Apache and PHP
- Then we'll talk about SSL
- some useful programs
- DNS



Redbrick
DCU's Networking Society

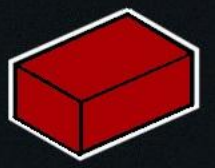
Securing apache

conf-available is where the modular configuration of apache happens (not related to domains/subdomains)

```
cd /etc/apache2/conf-available
```

Edit the security.conf file

```
sudo vim security.conf
```



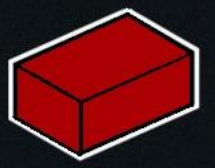
Redbrick

DCU's Networking Society

```
# Stop returning OS and version number
 ServerSignature Off
 ServerTokens Prod
```

Some other things to take look at later

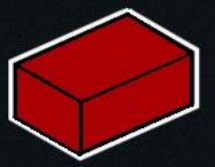
```
# Some optional things to think of including (I'd advise googling these, I'd only confuse you with my explanations)
 TimeOut
 KeepAliveTimeout
 LimitRequestBody
 LimitRequestFields
 LimitRequestFieldSize
 LimitRequestLine
 LimitXMLRequestBody
```



Redbrick

DCU's Networking Society

```
# Edit apache's core configuration
 cd /etc/apache2/
 sudo vim apache2.conf


# We'll add a domain name to the core of our server

 DocumentRoot /var/www/html

 <Directory /var/www/html>
   # Disable directory listings, following symbolic links and executing CGI scripts
   Options -Indexes -FollowSymLinks -ExecCGI
   # Don't let .htaccess files override configuration
   AllowOverride none
   # Let everyone access the files in /var/www/html through apache
   Require all granted
 </Directory>
```

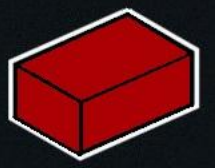


Redbrick

DCU's Networking Society

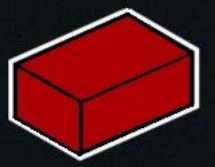
Securing PHP

```
# Open up the php config for apache
cd /etc/php5/apache2
sudo vim php.ini
```



Redbrick
DCU's Networking Society

```
; Restrict PHP Information Leakage (i.e. don't send PHP info in HTTP request headers)
expose_php = Off

; Disable Remote Code Execution
allow_url_fopen = Off
allow_url_include = Off ; This one should be off by default but make sure

; Disabling Dangerous PHP Functions - These are just a list of things we disable by default on our servers
disable_functions =
exec,passthru,shell_exec,system,proc_open,popen,curl_exec,curl_multi_exec,show_source,pcntl_alarm,pcntl_fork,pcntl_waitpid
,pcntl_wait,pcntl_wifexited,pcntl_wifstopped,pcntl_wifsignaled,pcntl_wexitstatus,pcntl_wtermsig,pcntl_wstopsig,pcntl_signal,pcntl
_signal_dispatch,pcntl_get_last_error,pcntl_strerror,pcntl_sigprocmask,pcntl_sigwaitinfo,pcntl_sigtimedwait,pcntl_exec,pcntl_get
priority,pcntl_setpriority

; Limit PHP execution to the following directories
;   i.e. Any PHP script anywhere else does not get executed, ever
open_basedir = "/var/www/html"

; Set some limits
upload_max_filesize = 2M  ; Maximum 2MB of file size a user can upload
max_execution_time = 30   ; 30 seconds execution time
max_input_time = 60       ; 60 seconds to parse POST or GET data from request
```



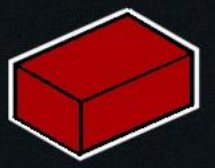
Redbrick

DCU's Networking Society

Some useful programs

```
sudo apt-get install fail2ban iotop htop iftop screen logwatch
```

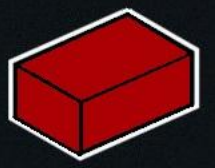
- **fail2ban**: Manages IP banning for your server. The settings out of the box are generally pretty awesome and deflect something like 5% of all Netsoc's requests.
- **iotop**: For tracking input/output
- **htop**: For tracking CPU processes and a good overview of the system (load on CPUs + memory)
- **iftop**: For tracking Network traffic
- **screen**: Used to create multiple terminal windows server-side. Great for leaving unnattended/long-running programs going (EG: weechat or some IRC client)
- **logwatch**: Processes your logs and will email you some of the highlights and things you should look out for. You should have a look through the config for yourself, it's different for everyone. What you will have to do is create a cronjob for this to run and email you every so often. (we run it nightly)



Redbrick
DCU's Networking Society

LetsEncrypt: Free SSL for everyone!

https://letsencrypt.org/ < find all the docs you want here



Redbrick
DCU's Networking Society

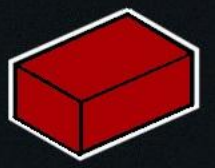
IP addresses are ugly, so…?

You will need to buy a domain or get a free one from one of the links below!

http://www.freenom.com/en/index.html?lang=en

Namecheap.com

The github education pack gives you a free DNS provider too!



Redbrick

DCU's Networking Society