AWS/LAMP/Wordpress/Bootstrap Extravaganza

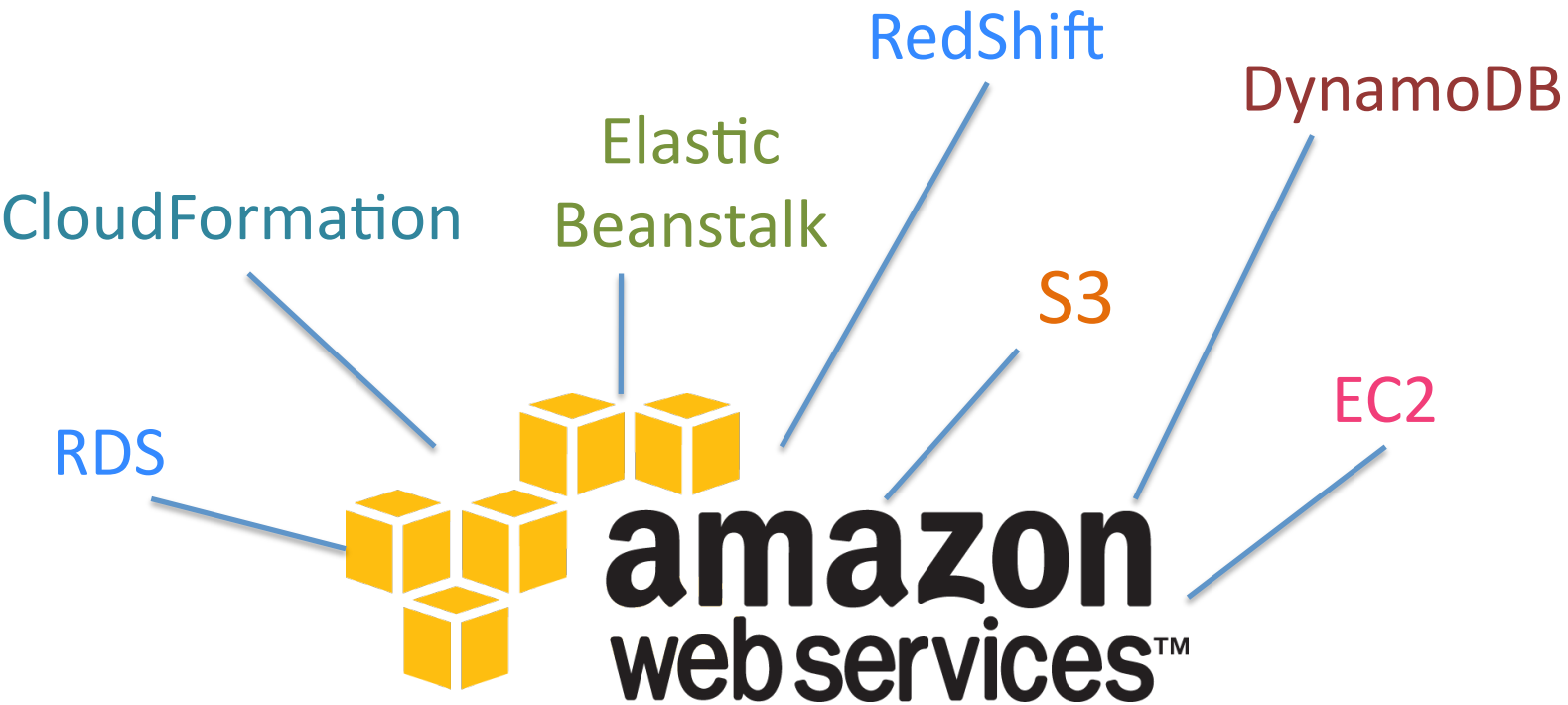
I am determined to make this shit fun.

Agenda

- What is AWS?
- Launch your instance
- Deploy LAMP stack
- Install Wordpress
- Bootstrap it
- Q&A

What is AWS?

- Amazon Web Services
- (It's awesome)



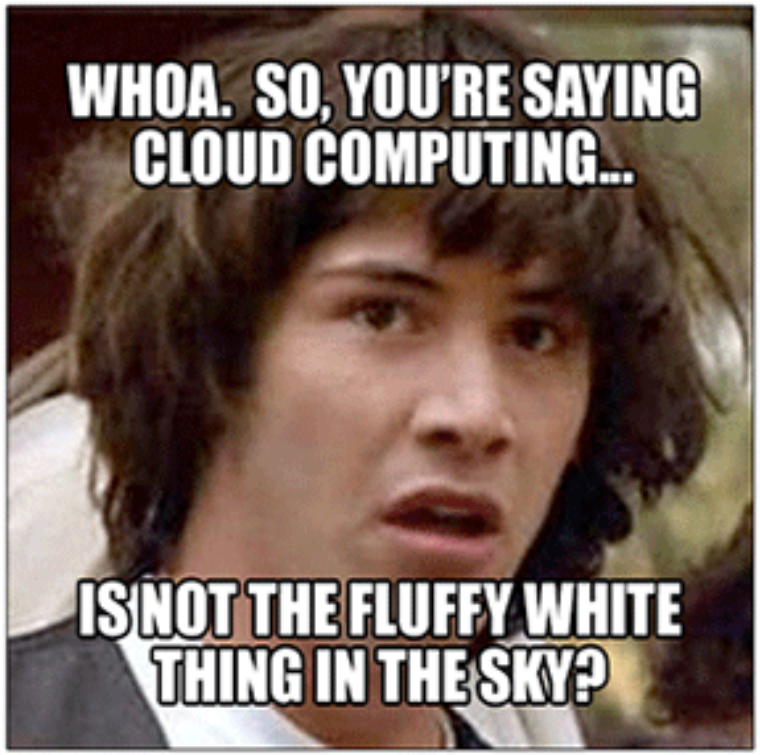
What service will you be using?

- Amazon EC2
  - Elastic Cloud Compute

You will be building

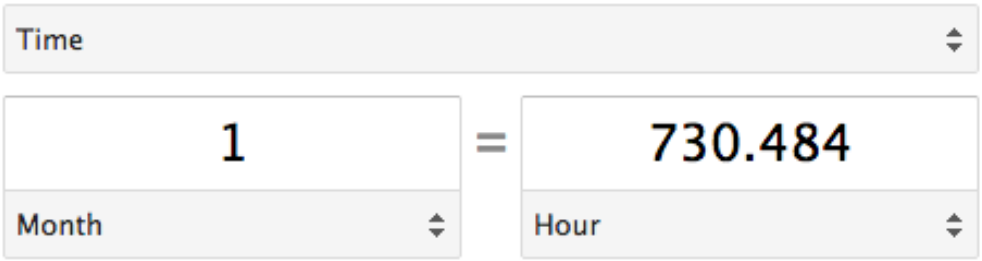
your very own!

Hooray!

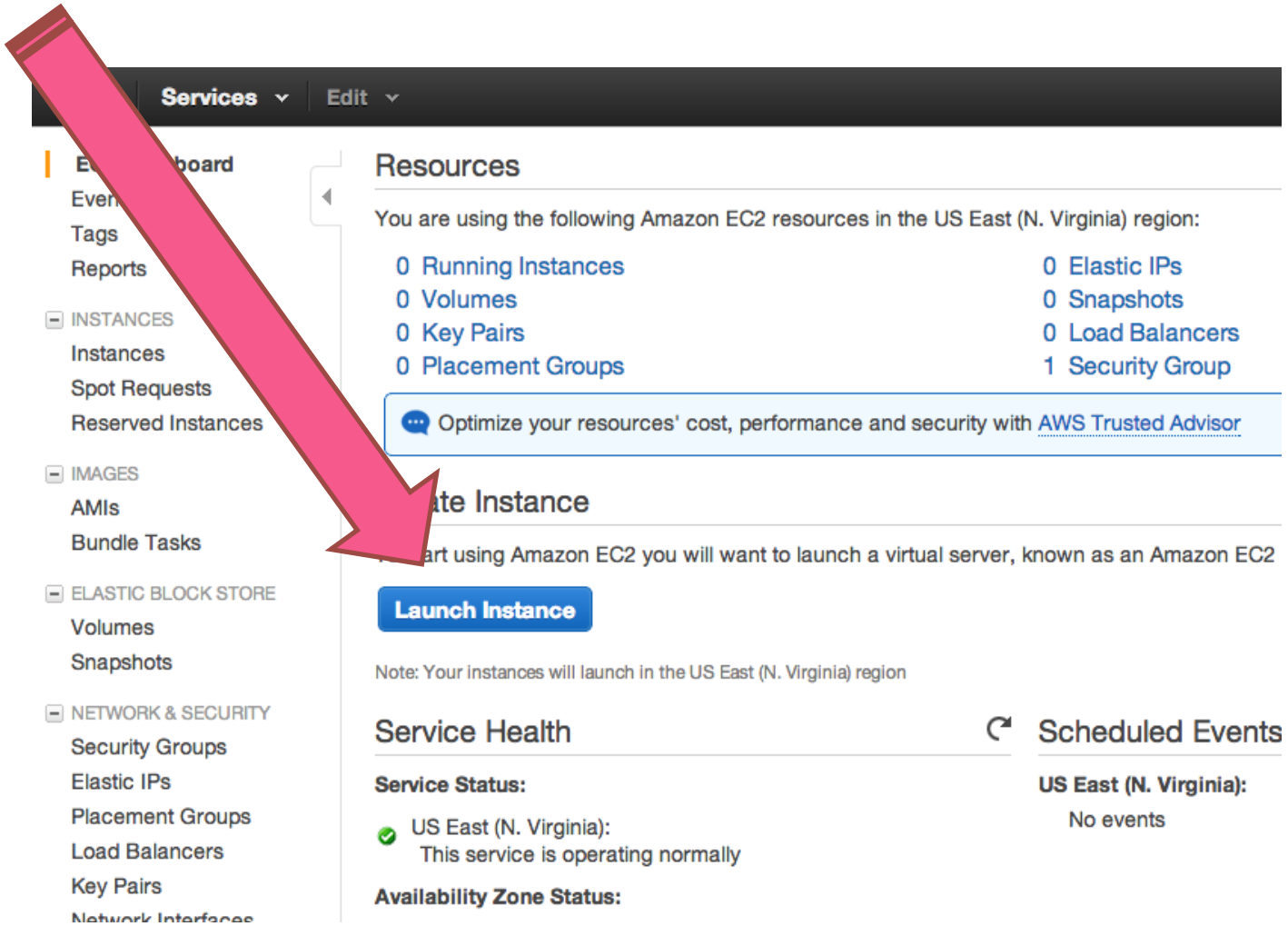


Let’s get started, yo

- Create your AWS account
  - http://aws.amazon.com
  - You will need:
    - Your phone
    - A credit card (but never fear!)
      - AWS EC2 Free Tier Usage Limitations
      - http://aws.amazon.com/free/

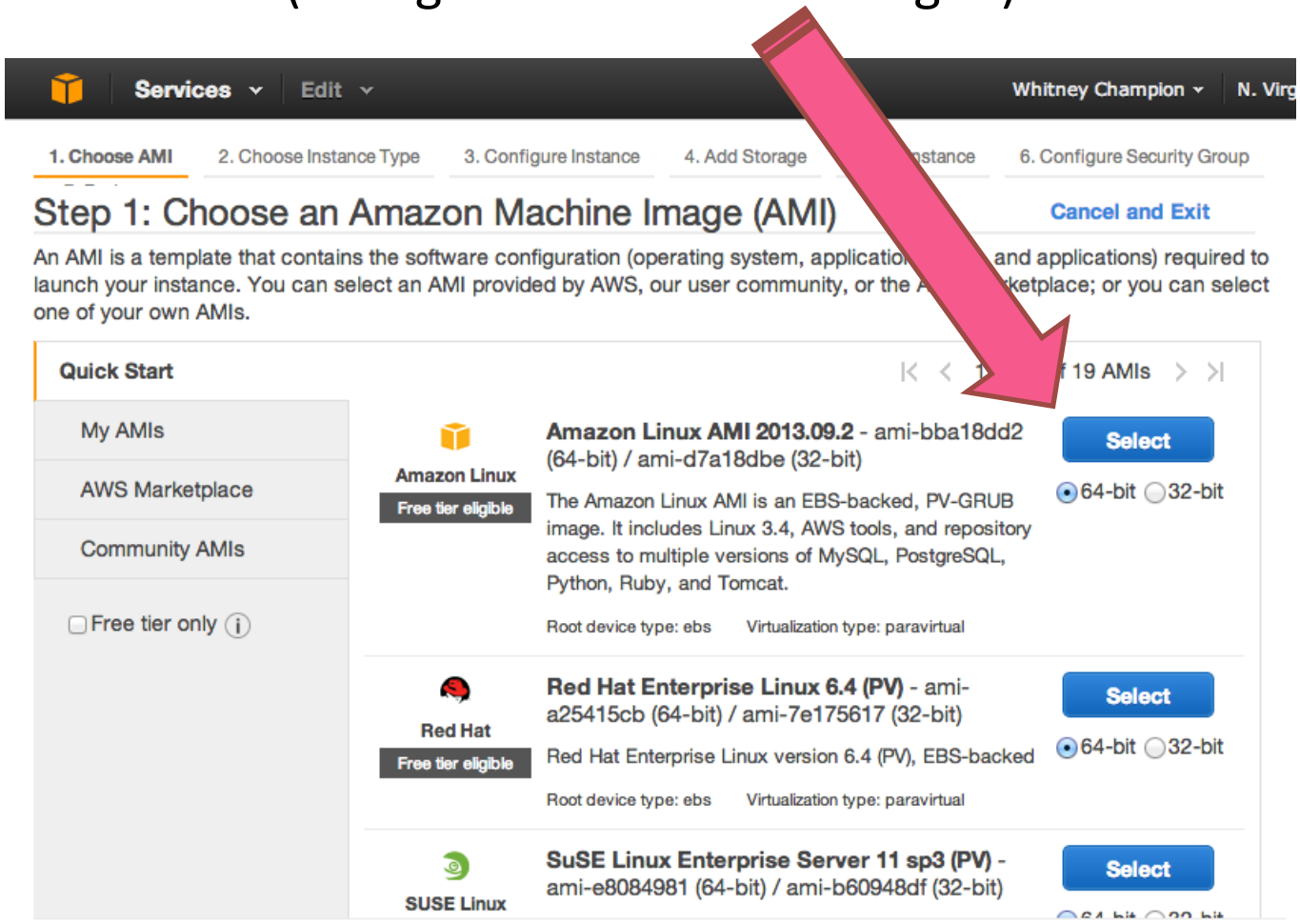


Launch your instance

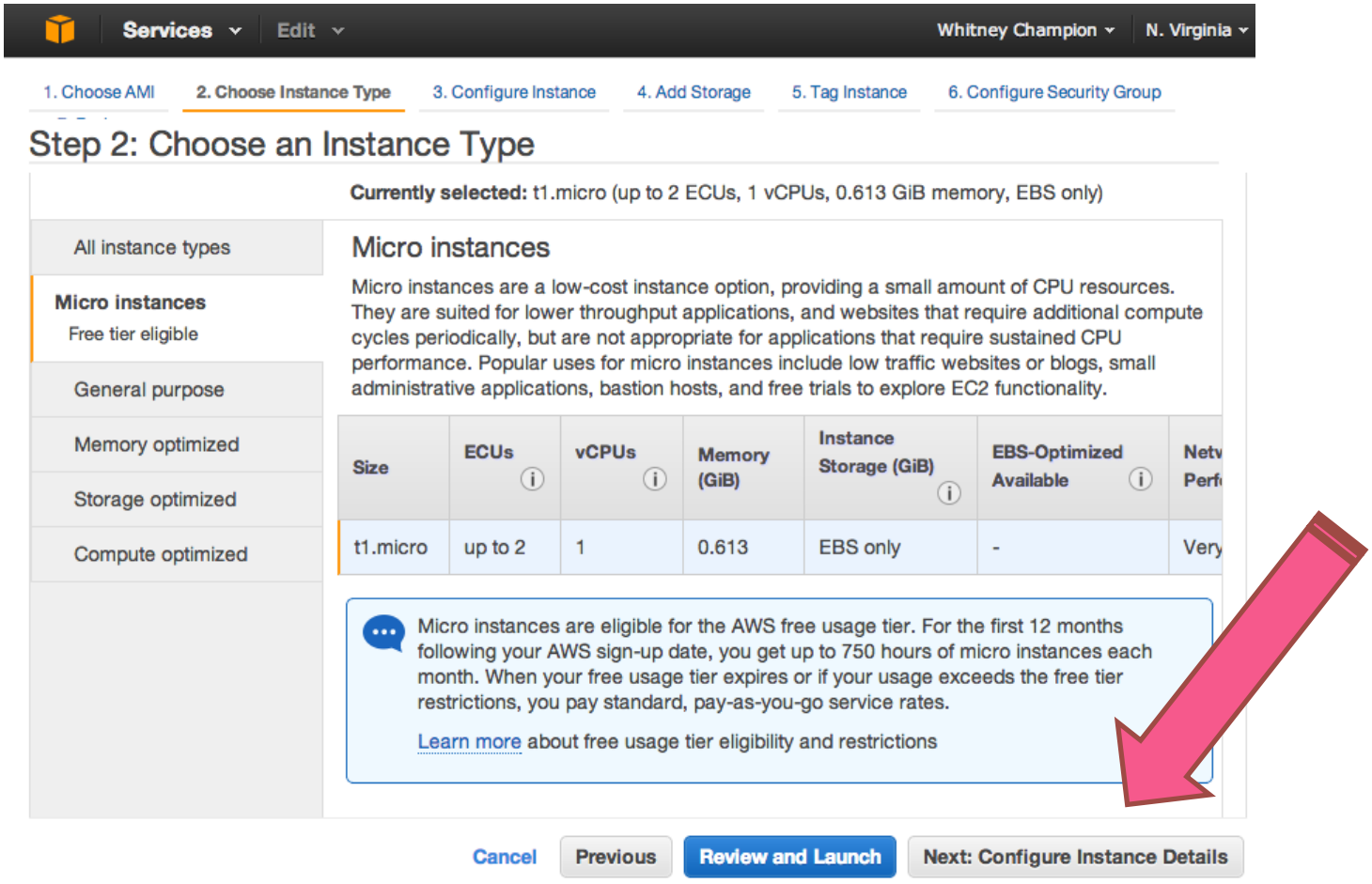


Pick your flavor

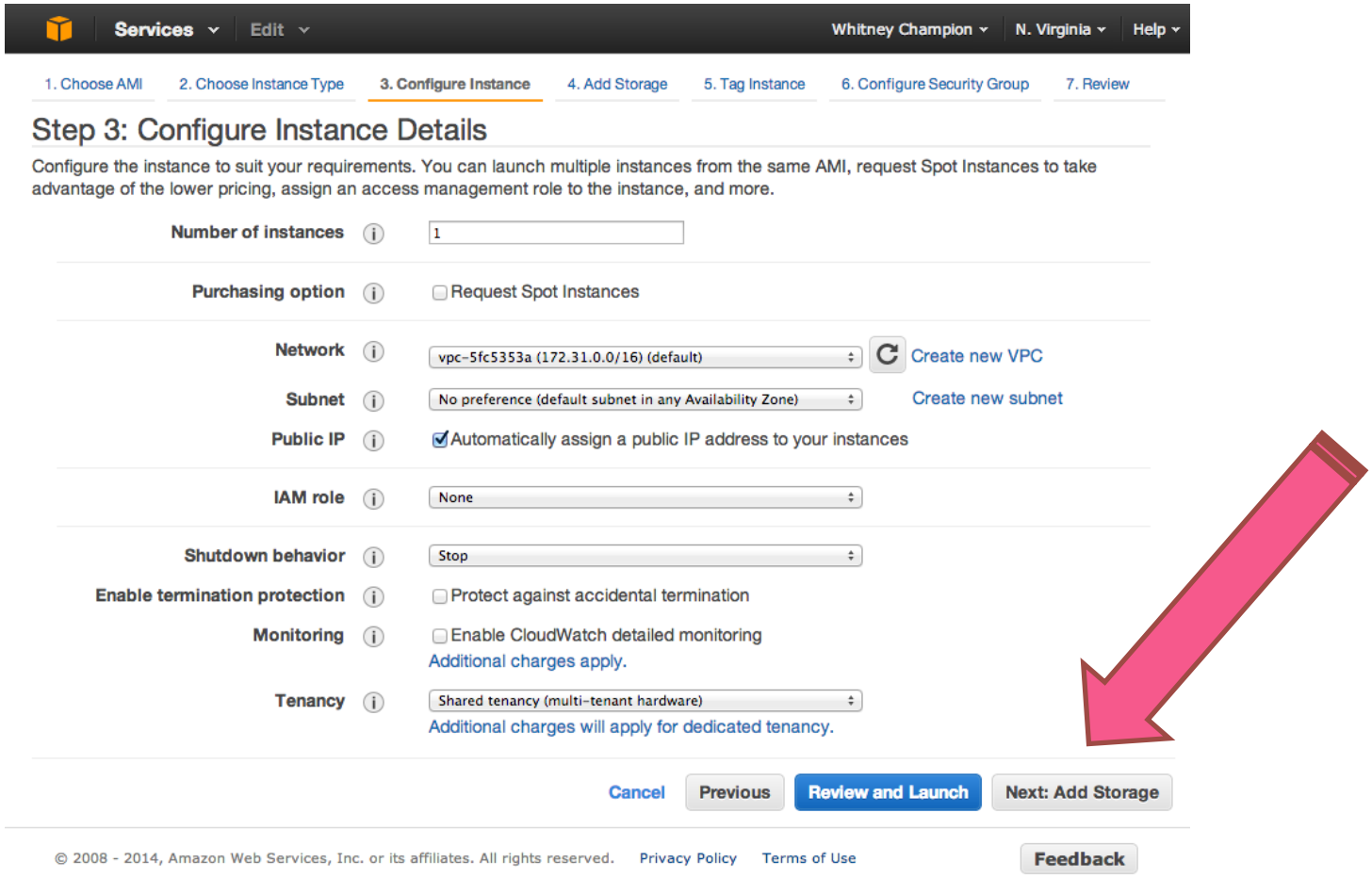
(*cough* RHEL based *cough*)



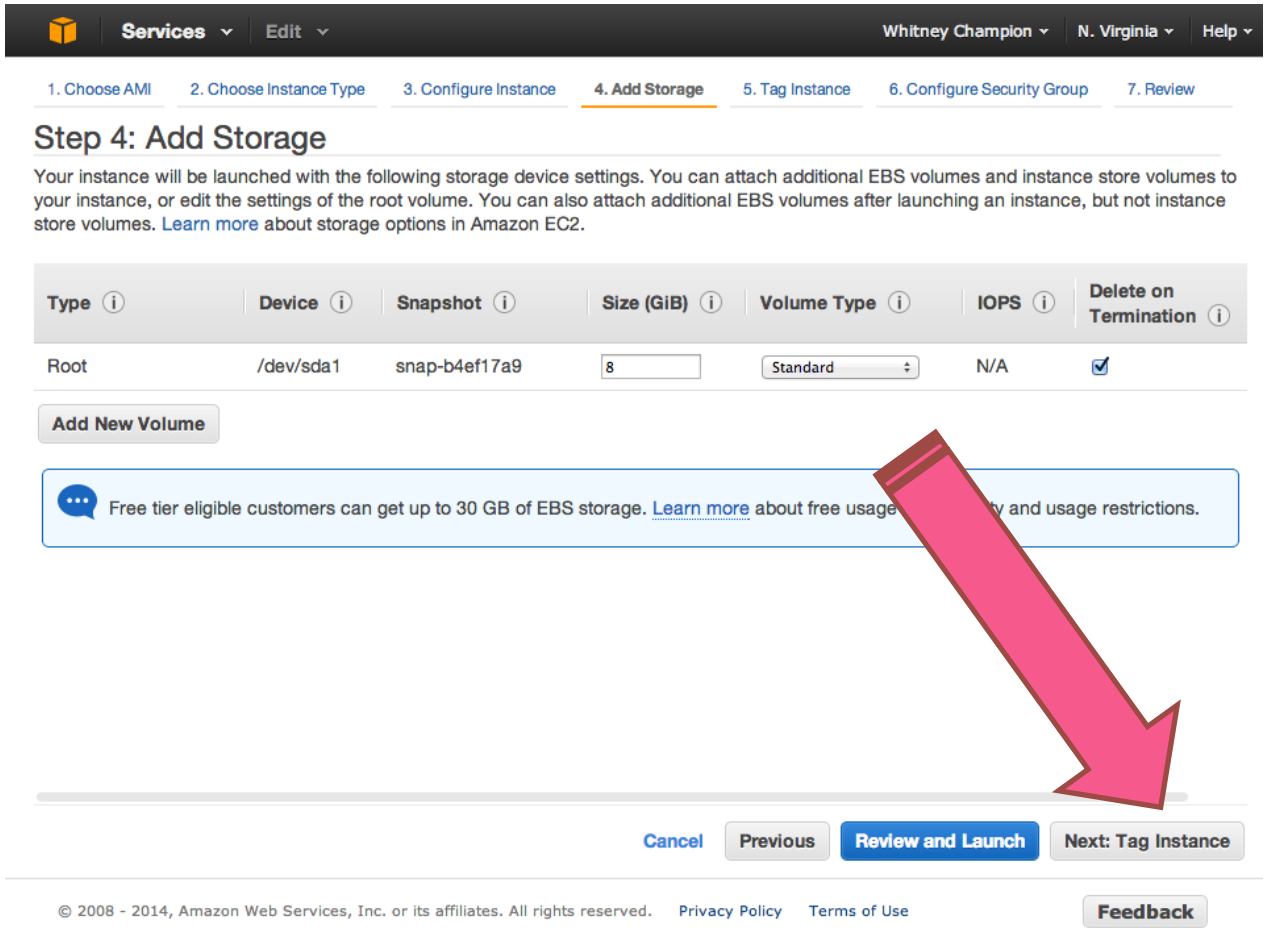
Pick ~~a size,~~ Micro because it's free



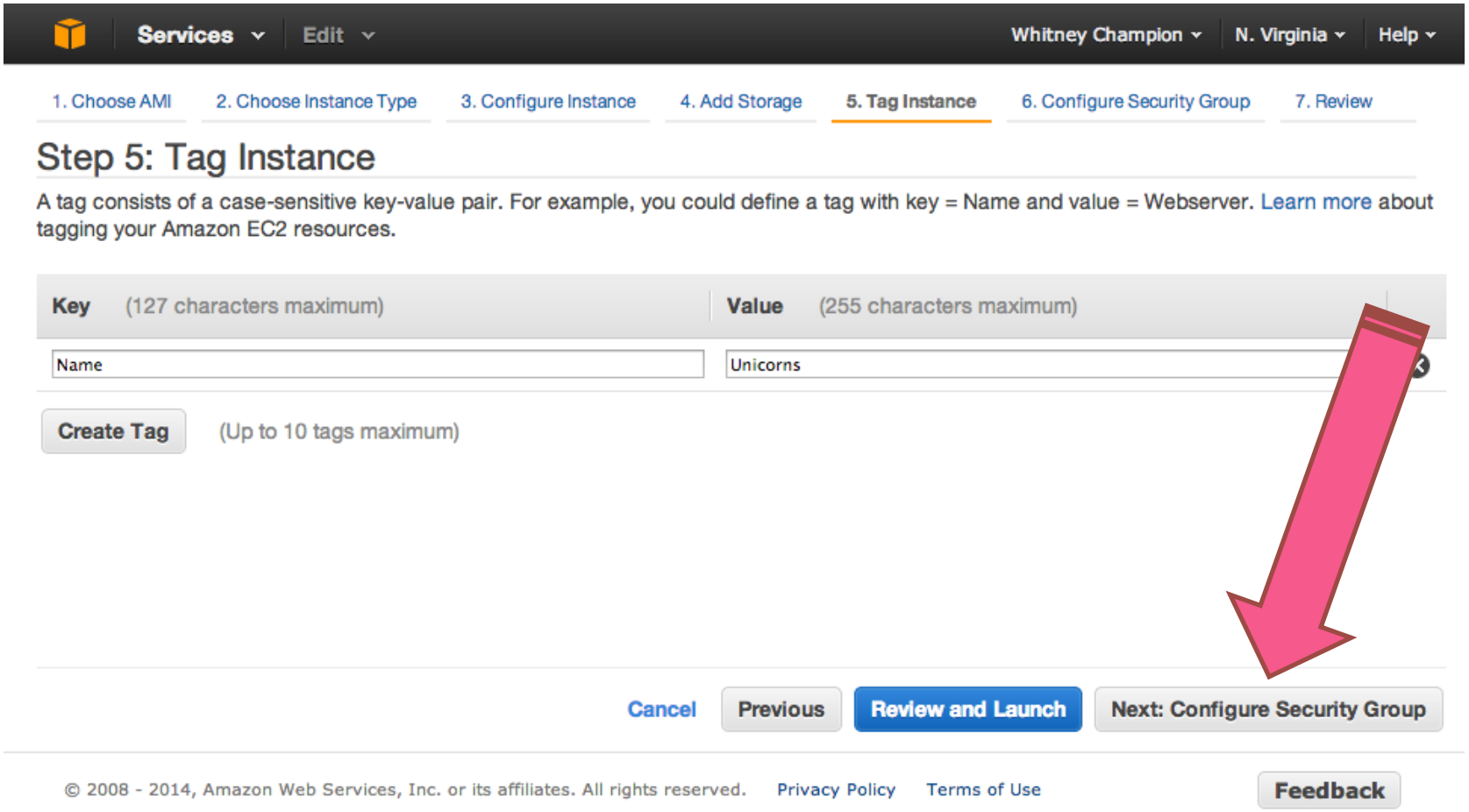
Configure details



Add storage


Services
Edit
Whitney Champion
N. Virginia
Help
1. Choose AMI
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Tag Instance
6. Configure Security Group
7. Review
Step 4: Add Storage
Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. Learn more about storage options in Amazon EC2.
Type
Device
Snapshot
Size (GiB)
Volume Type
IOPS
Delete on Termination
Root
/dev/sda1
snap-b4ef17a9
8
Standard
N/A
Add New Volume
Free tier eligible customers can get up to 30 GB of EBS storage. Learn more about free usage
and usage restrictions.
Cancel
Previous
Review and Launch
Next: Tag Instance
© 2008 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use
Feedback

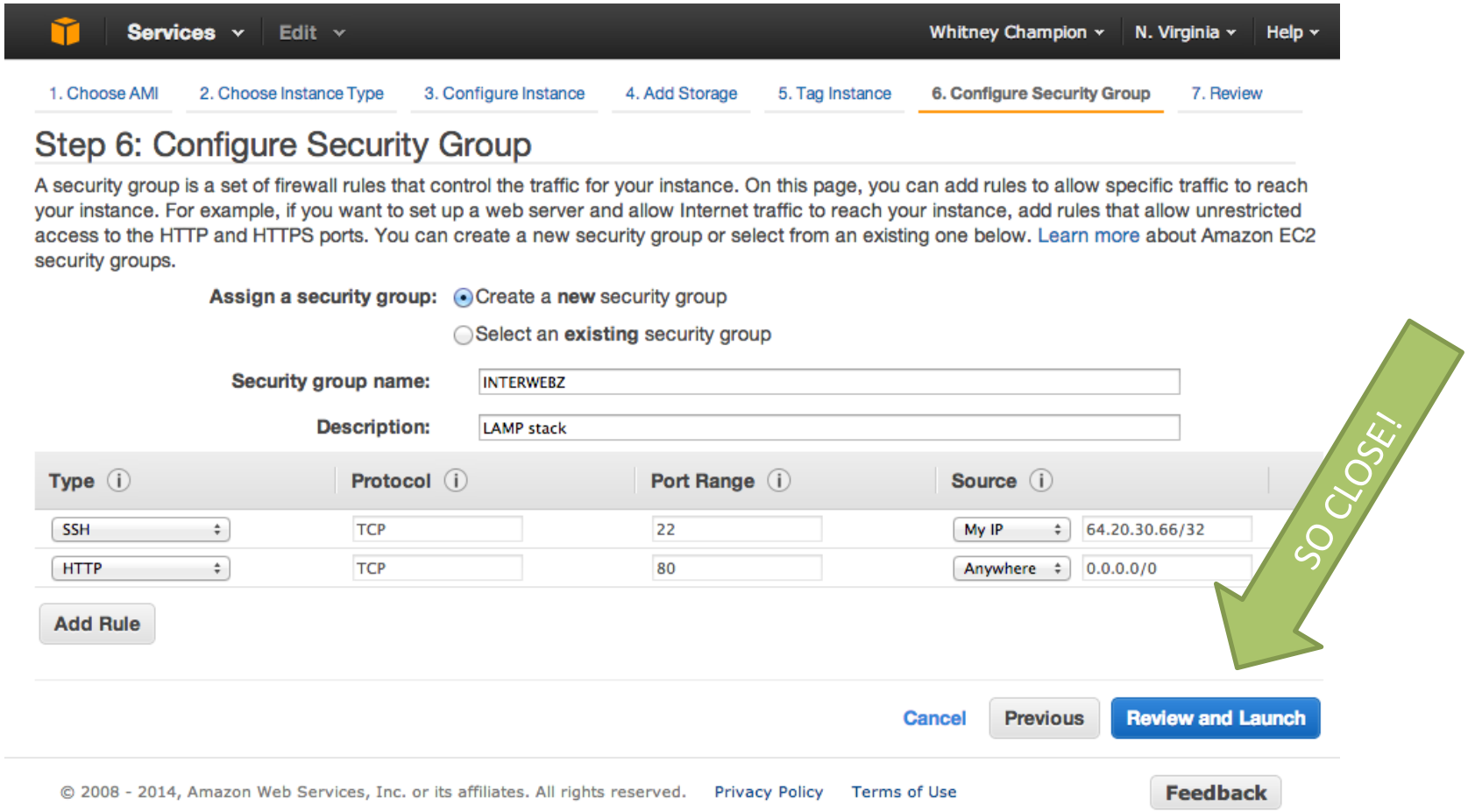
Name all the things!



Configure security group

- But… what is a security group?
  - Says what traffic can get to your instance on what ports
- What ports should we have open?
  - 80 – Apache web server
  - 22 – SSH access

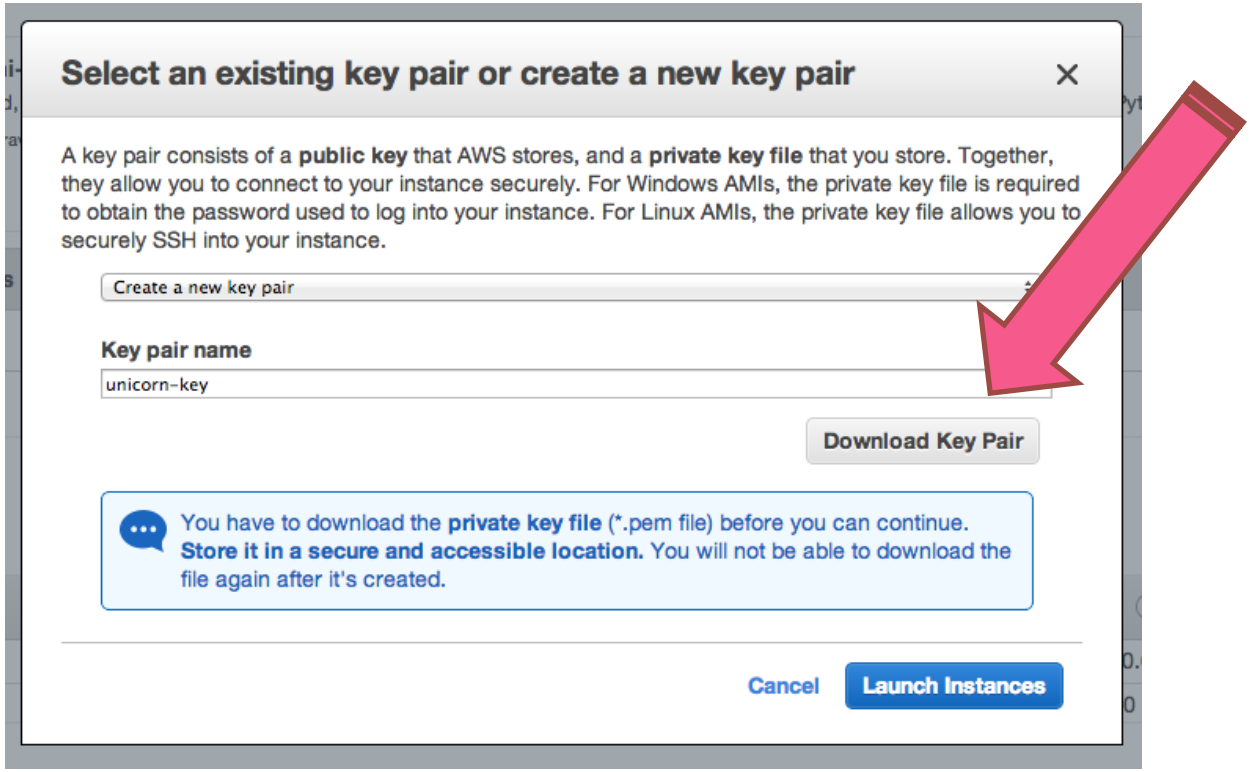
Configure security group



Review and…

Except not really!

- j/k!
- You have to have a private key to connect to your instance

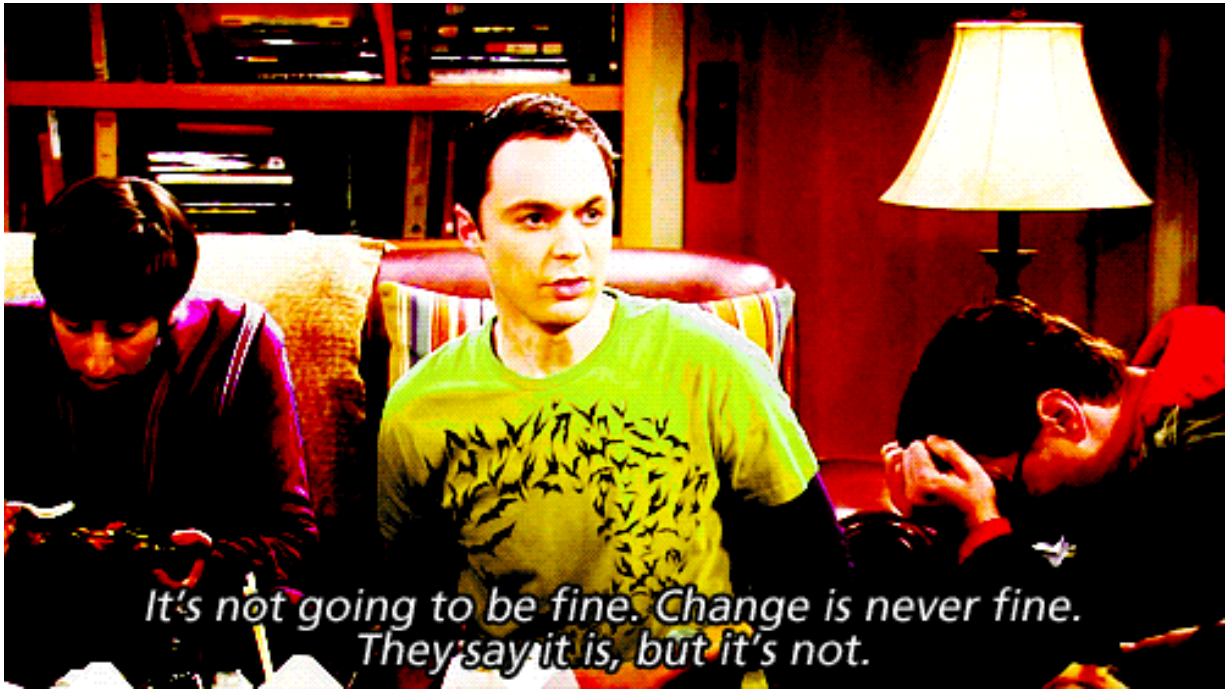


Now you can launch

- For real this time

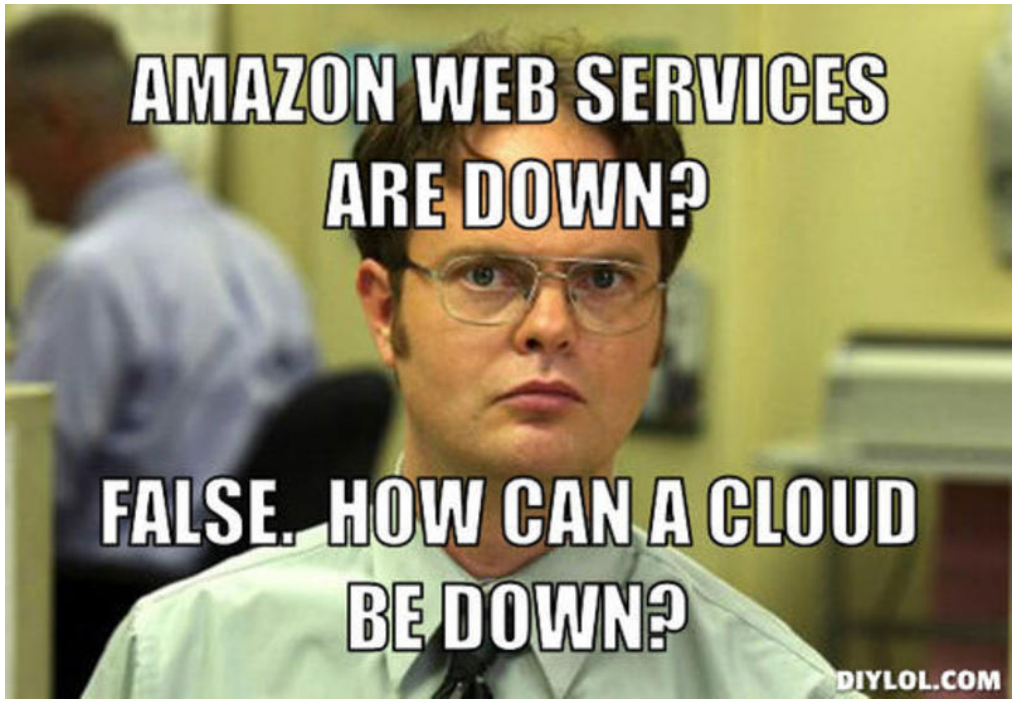
Get an Elastic IP

- Public IPs change on reboot
- Elastic IPs do not



Monitoring! It's important

- Elastic load balancers double as port monitoring



Connect to your instance

- Windows options
  - PuTTY
  - Cygwin

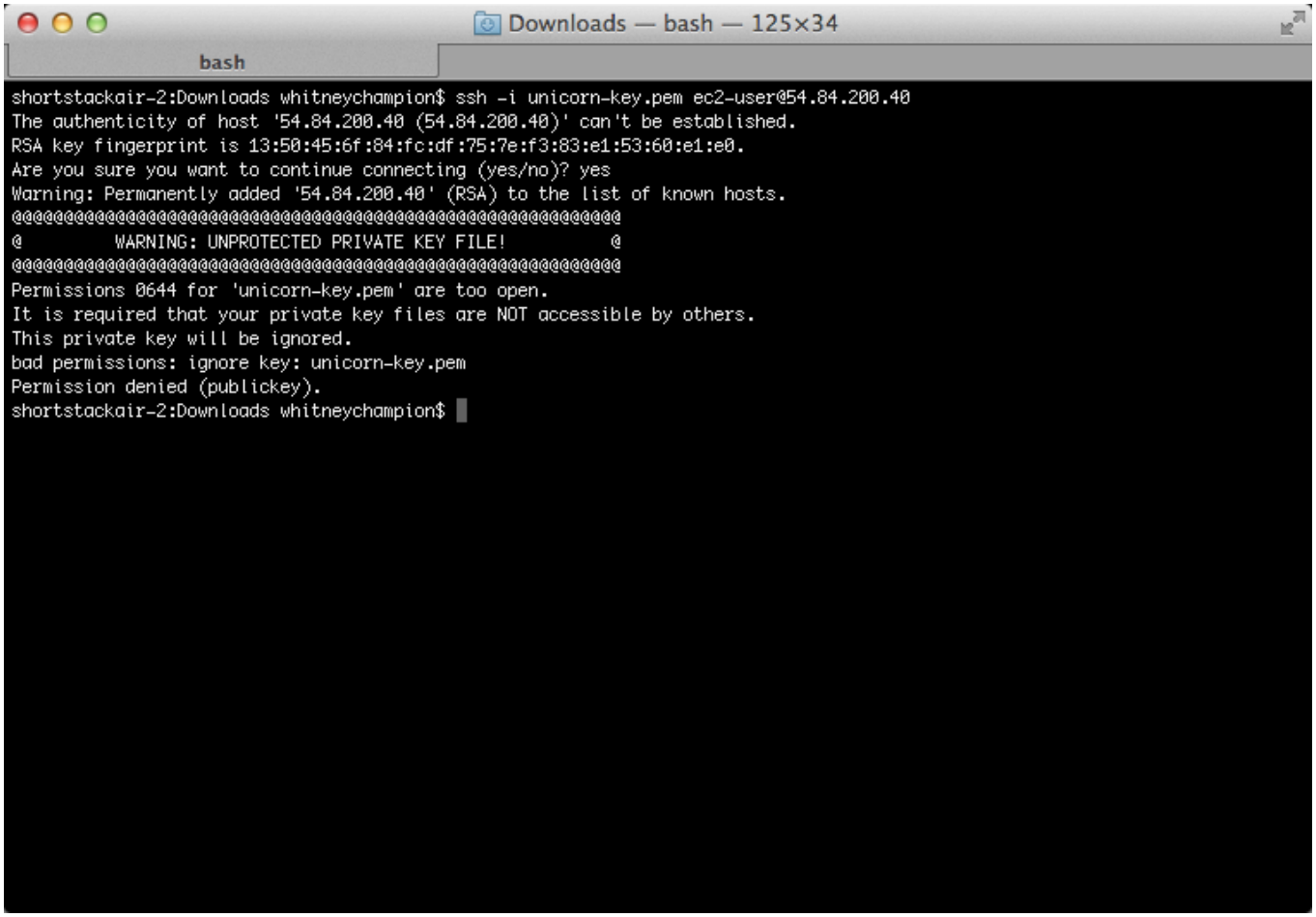
- Mac - Terminal

```
ssh -i /Users/{your-username}/Downloads/{your-private-
        key.pem} ec2-user@{your-public-ip}
```

- Linux- Terminal

```
ssh -i /home/{your-username}/Downloads/{your-private-
        key.pem} ec2-user@{your-public-ip}
```

Oh noes!


Downloads — bash — 125×34
bash
shortstackair-2:Downloads whitneychampion$ ssh -i unicorn-key.pem ec2-user@54.84.200.40
The authenticity of host '54.84.200.40 (54.84.200.40)' can't be established.
RSA key fingerprint is 13:50:45:6f:84:fc:df:75:7e:f3:83:e1:53:60:e1:e0.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '54.84.200.40' (RSA) to the list of known hosts.
@@@
@ WARNING: UNPROTECTED PRIVATE KEY FILE! @
@@@
Permissions 0644 for 'unicorn-key.pem' are too open.
It is required that your private key files are NOT accessible by others.
This private key will be ignored.
bad permissions: ignore key: unicorn-key.pem
Permission denied (publickey).
shortstackair-2:Downloads whitneychampion$

chmod ftw

- Private keys need to have permissions of 600 or lower
- Based on 3 binary numbers

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1
					r	w	x

	r	w	x	Total
Owner	4	2	0	**6**
Group	0	0	0	**0**
Other	0	0	0	**0**

```
chmod 600 {your-private-key.pem}
```

Now connect to your instance

- Windows options
  - PuTTY
  - Cygwin

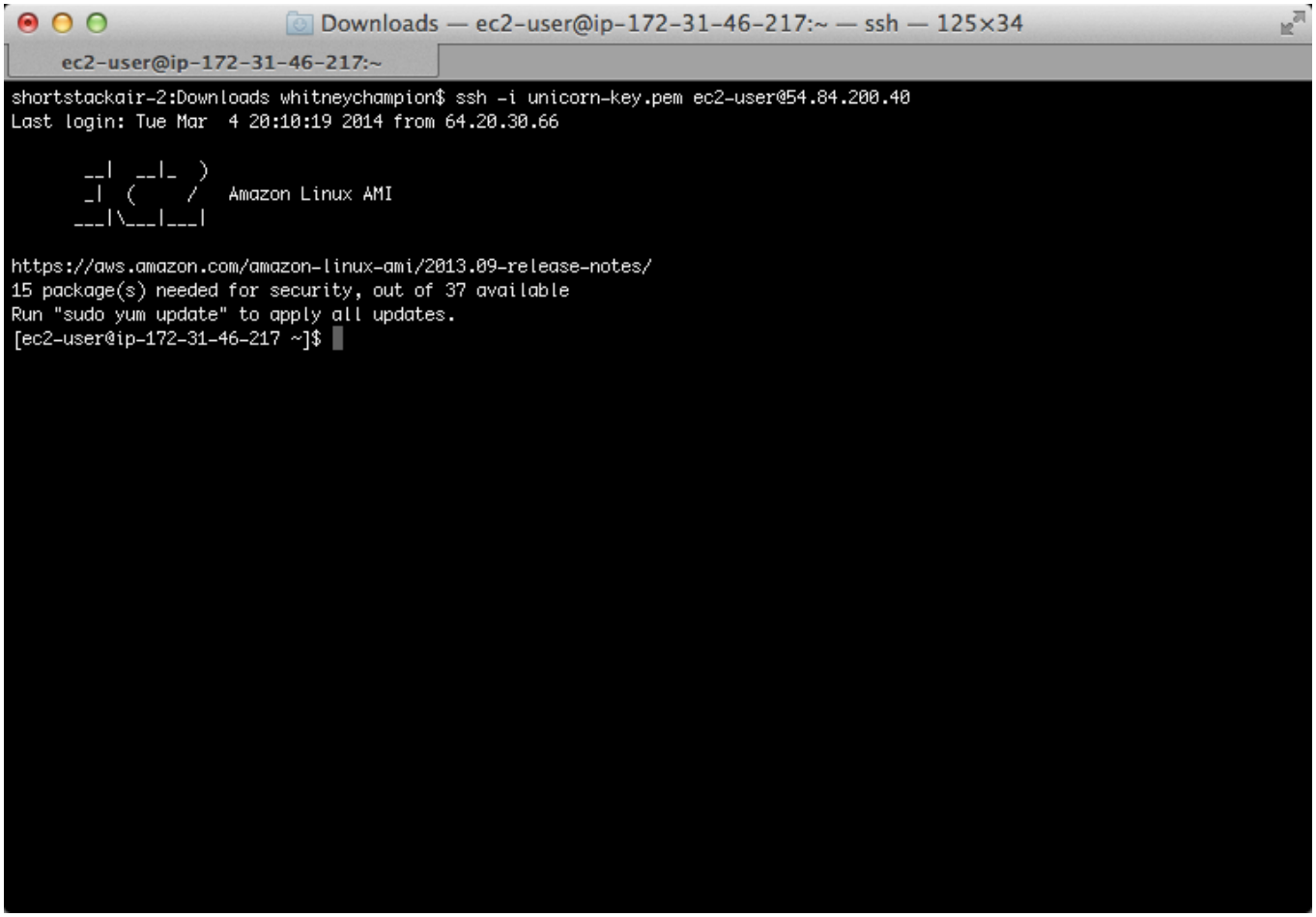
- Mac - Terminal

```
ssh -i /Users/{your-username}/Downloads/{your-private-
        key.pem} ec2-user@{your-public-ip}
```

- Linux- Terminal

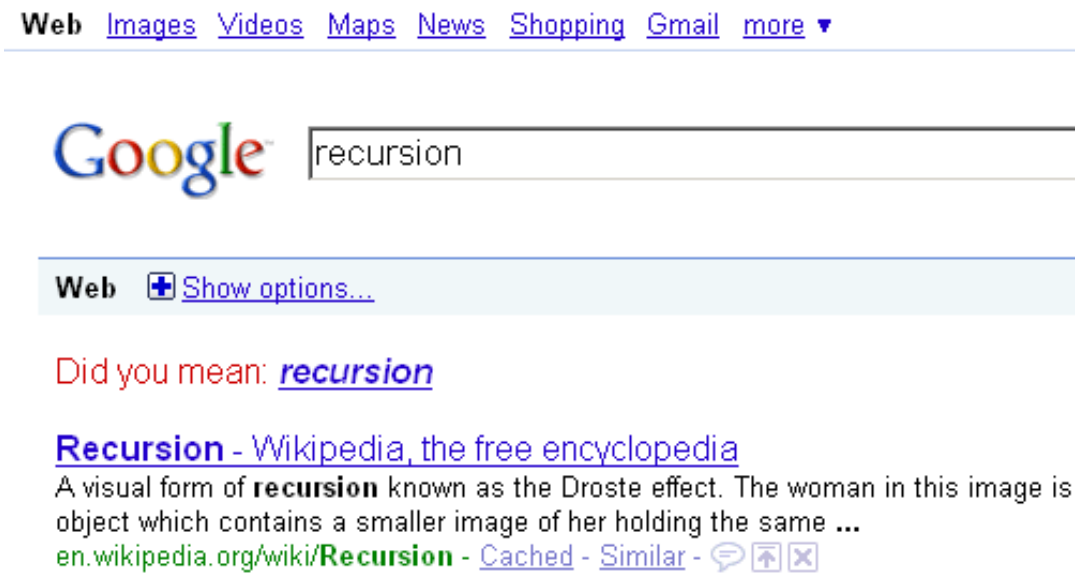
```
ssh -i /home/{your-username}/Downloads/{your-private-
        key.pem} ec2-user@{your-public-ip}
```

Success!


Downloads — ec2-user@ip-172-31-46-217:~ — ssh — 125×34
ec2-user@ip-172-31-46-217:~
shortstackair-2:Downloads whitneychampion$ ssh -i unicorn-key.pem ec2-user@54.84.200.40
Last login: Tue Mar 4 20:10:19 2014 from 64.20.30.66
__| __|_)
_| (/ Amazon Linux AMI
___|___|___|
https://aws.amazon.com/amazon-linux-ami/2013.09-release-notes/
15 package(s) needed for security, out of 37 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-46-217 ~]$

Updates and security patches

- What is yum?
  - Yellowdog Updater, Modified
- What is an RPM?
  - Red Hat Package Manager
  - RPM Package Manager



Update your system

```
$ sudo yum update
```

Easy, right? Right!

Now we wait


THE #1 PROGRAMMER EXCUSE FOR LEGITIMATELY SLACKING OFF:
"MY CODE'S COMPILING."
HEY! GET BACK TO WORK!
COMPILING!
OH. CARRY ON.

LAMP Stack

- ✓ Linux – Done!
- Apache – Let's do dis…
- MySQL
- PHP

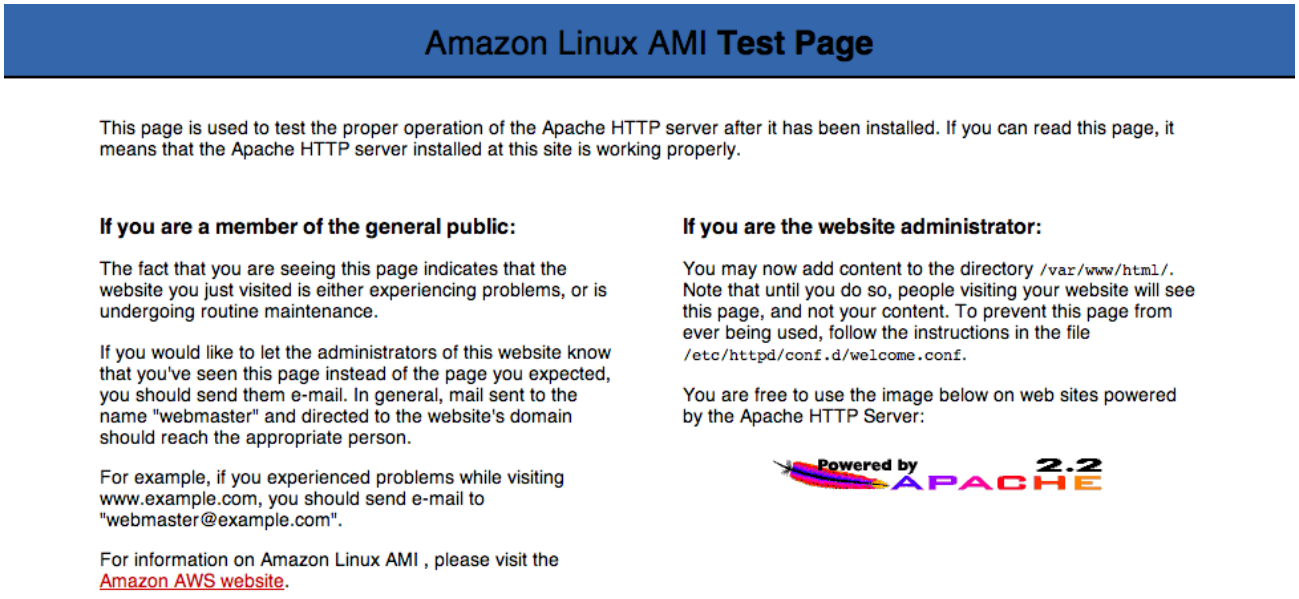
Apache

```
$ sudo yum install httpd
$ sudo service httpd start
$ sudo chkconfig httpd on
```

- What did we just do?
  - Installed Apache
  - Started the web server
  - Made sure Apache was set to start on boot

Check ‘er out

- Browser → http://{your-public-ip}



Amazon Linux AMI **Test Page**

This page is used to test the proper operation of the Apache HTTP server after it has been installed. If you can read this page, it means that the Apache HTTP server installed at this site is working properly.

If you are a member of the general public:

The fact that you are seeing this page indicates that the website you just visited is either experiencing problems, or is undergoing routine maintenance.

If you would like to let the administrators of this website know that you've seen this page instead of the page you expected, you should send them e-mail. In general, mail sent to the name "webmaster" and directed to the website's domain should reach the appropriate person.

For example, if you experienced problems while visiting www.example.com, you should send e-mail to "webmaster@example.com".

For information on Amazon Linux AMI , please visit the Amazon AWS website.

If you are the website administrator:

You may now add content to the directory `/var/www/html/`. Note that until you do so, people visiting your website will see this page, and not your content. To prevent this page from ever being used, follow the instructions in the file `/etc/httpd/conf.d/welcome.conf`.

You are free to use the image below on web sites powered by the Apache HTTP Server:

Powered by APACHE 2.2

Questions so far?

LAMP Stack

- ✓ Linux – Done!
- ✓ Apache – Done!
- MySQL – Onward!
- PHP – Let's knock this one out, too!

Let’s first check our resources…

```
$ free
```

Create a swap file

```
$ sudo dd if=/dev/zero of=/swapfile bs=1M count=1024
$ sudo chmod 600 /swapfile
$ sudo mkswap /swapfile
$ sudo swapon /swapfile
$ sudo vim /etc/fstab
   /swapfile swap swap defaults 0 0
$ sudo mount -a
```

- What did we just do?
  - Made a 1G file full of 0's
  - Made it accessible only to root
  - Turned that file into swap space
  - Enabled our new swap space
  - Enabled on reboot
  - Tested our changes (always… always do this)

Install MySQL & PHP

```
$ sudo yum install mysql mysql-server php php-xml php-mysql
$ sudo service mysqld start
$ sudo chkconfig mysqld on
$ sudo service httpd restart
```

- What did we just do?
  - Installed MySQL command line tools
  - Installed MySQL server
  - Installed PHP and PHP extensions
  - Started MySQL server, and set it to start on boot
  - Restarted Apache, because it needs to know PHP is now installed and available

Configure MySQL

```
$ sudo /usr/bin/mysql_secure_installation

Enter current password for root (enter for none):
Set root password? [Y/n] Y
Remove anonymous users? [Y/n] Y
Disallow root login remotely? [Y/n] Y
Remove test database and access to it? [Y/n] Y
Reload privilege tables now? [Y/n] Y
```

MAKE A NOTE OF ROOT PASSWORD

Configure Wordpress database

```
$ mysql –u root –p

mysql> show databases;
mysql> create database wordpress;
mysql> grant usage on *.* to wpadmin@localhost
  identified by 'googlerules';
mysql> grant all privileges on wordpress.* to
  wpadmin@localhost;
```

MAKE A NOTE OF YOUR CREDENTIALS

Install phpMyAdmin

- What is phpMyAdmin?
  - A MySQL administration tool

```
$ wget http://packages.sw.be/rpmforge-release/rpmforge-
   release-0.5.2-2.el5.rf.i386.rpm
$ sudo rpm -Uvh rpmforge-release-0.5.2-2.el5.rf.i386.rpm
$ sudo yum install phpmyadmin
```

Configure phpMyAdmin

```
$ sudo vim /etc/httpd/conf.d/phpmyadmin.conf
   Allow from 64.20.30.66

$ sudo vim /etc/httpd/conf/httpd.conf
   <Directory /var/www/html>
      AllowOverride All

$ sudo vim /usr/share/phpmyadmin/config.inc.php
   $cfg['blowfish_secret'] = 'magical-unicorns';

$ sudo service httpd reload
```

Check ‘er out

- Browser →

  http://{your-public-ip}/phpMyAdmin

- Log in with your Wordpress MySQL username and password

LAMP Stack

- ✓ Linux – Done!
- ✓ Apache – Done!
- ✓ MySQL – Done!
- ✓ PHP – Done!

- ✓ *Plus* you're ready for Wordpress and have a sweet MySQL GUI

Let's re-cap…

So far, you have built…

- An EC2 instance in the AWS cloud with:
  - Amazon's flavor of Linux
  - An Apache web server
  - A swap partition
  - A MySQL server with a Wordpress-ready database

Onward!

Let’s revisit port monitoring

Time for Wordpress

```
$ cd /var/www/html
$ sudo wget http://wordpress.org/latest.tar.gz
$ sudo tar -xvf latest.tar.gz --strip-components 1
$ sudo chown –R apache:apache /var/www/html
```

- What did we just do?
  - Downloaded Wordpress
  - Unzipped it to the proper directory so it will show up on our web server
  - Made sure all Wordpress files are owned by Apache

Install Wordpress

- Browser → http://{your-public-ip}/wp-admin/install.php

Twitter Bootstrap

- What is bootstrap?

Get some plugins

- Browser →http://{your-public-ip}/wp-admin
- Plugins → Add New
  - Bootstrap Shortcodes
  - CPT Bootstrap Carousel
- Install and activate
- Shiny plug-and-play goodness awaits

Let’s make it pretty!

- Appearance → Themes → Add New

Tinker

- Make a carousel and embed it in a page
  - [image-carousel]
- Make your homepage a static page instead of blog posts
- Add a few pages and menu items
- Test out the shortcodes
- Add some homepage widgets
  - Hint: You can use the sweet bootstrap shortcodes inside a “Text” widget
- Play with the CSS inside of the theme editor

Now you have…

- A server
- A website
- A responsive website
- **AND IT'S FREE FOR A YEAR.**
- **AND YOU BUILT IT.**
- **HELLO FREE TRAINING FOR A YEAR.**

Profit!

Here’s to choosing the hard way :)

Questions?