Ziyun He
Steven Louie

CMPE 172 Assignment #1

Assignment:

1. Configure Ansible to deploy web server, and bring it up a port 80 with a web page that is publicly accessible that displays the message: “Hello World”.
2. Include in the Ansible playbook, plays to deploy and undeploy the resources

Steps:

1. Install Ansible
2. Launch EC2 instance
3. Configure SSH
4. Configure Ansible
5. Create an HTML File that Displays “Hello World”
6. Create a playbook that deploys a web server
7. Deploy playbook
8. Create a playbook that un-deploys a web server
9. Un-deploy playbook

Install Ansible:

1. Go to offical website: https://docs.ansible.com/ansible/latest/installation_guide/intro_installation.html
2. Follow installation guide to install Ansible on Mac, using pip

Ansible can be installed via “pip”, the Python package manager.

```
$ sudo easy_install pip
```

Then install Ansible with [1]:

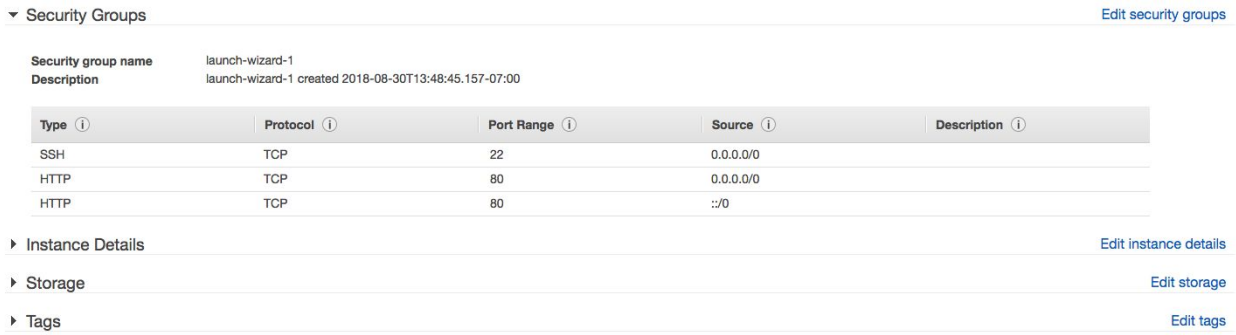
```
$ sudo pip install ansible
```

3. Verify if Ansible is successfully installed, using “ansible --version” in terminal

```
Ziyuns-MacBook-Air:~ ziyun$ ansible --version
ansible 2.6.3
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/Users/ziyun/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  ansible python module location = /Library/Python/2.7/site-packages/ansible
  executable location = /usr/local/bin/ansible
  python version = 2.7.10 (default, Oct  6 2017, 22:29:07) [GCC 4.2.1 Compatible Apple LLVM 9.0.0 (clang-900.0.31)]
```

Launching EC2 Instance on AWS and Enabling Port 80

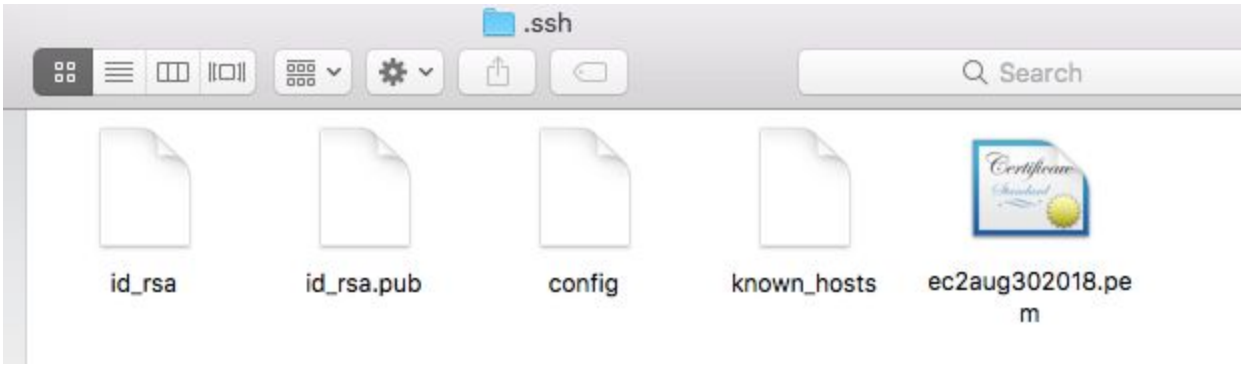
1. Go to https://aws.amazon.com/ec2/ and create an account. After creating an account, head to EC2 service.
2. In the EC2 dashboard, we launched an instance and chose Ubuntu Server 16.04 LTS as our operating system. From there we launched our instance with default settings.
3. After creating our EC2 instance, we will now enable port 80 for http requests by going to Security Groups located on the left-hand side of the EC2 Management Console.
4. From there we edited the inbound rules and added a rule to enable port 80 for http requests.



5. Download the key-pair (.pem file)

Configure SSH

1. To create a ssh key pair on the server machine, run this command in terminal:
   ssh-keygen -t rsa
2. That command will then prompt a few questions, which we want to just hit enter on:
   a. Enter file in which to save the key(~/.ssh/id_rsa):
   b. Enter passphrase (empty for no passphrase):
3. When launching EC2 instance, AWS will provide us a private key pair in this case, which would later be needed to access our EC2 instance from our machine.
4. Put pem file in ssh folder



5. To verify that we can establish a connection via ssh to our EC2 instance from our machine, we run the command:
   ssh -i /Users/ziyun/.ssh/ec2aug302018.pem ubuntu@54.215.144.152

```
Ziyuns-MacBook-Air:~ ziyun$ ssh -i /Users/ziyun/.ssh/ec2aug302018.pem ubuntu@54.215.144.152
Welcome to Ubuntu 16.04.5 LTS (GNU/Linux 4.4.0-1065-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  Get cloud support with Ubuntu Advantage Cloud Guest:
    http://www.ubuntu.com/business/services/cloud

2 packages can be updated.
0 updates are security updates.

New release '18.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.


*** System restart required ***
Last login: Fri Aug 31 21:20:24 2018 from 130.65.254.18
ubuntu@ip-172-31-1-159:~$
```

6. After connect to EC2 instance, install python on it, using:
   sudo apt-get install python-simplejson

Configure Ansible

1. After installing ansible, we created the list of client machines we wish to access this server by running the below command
   Sudo vi /etc/ansible/hosts
2. In our hosts file, head all the way to the bottom of the file and add the following:

```
hosts
44
45  [ec2]
46  54.215.144.152 ansible_ssh_user=ubuntu ansible_ssh_private_key_file=/Users/ziyun/.ssh/ec2aug302018.pem
47
```

3. To verify that we can reach the client machine, run the following command:
    a. Ansible -m ping all

   This should output ping result success as shown below:

```
Ziyuns-MacBook-Air:ansible ziyun$ ansible -m ping all
54.215.144.152 | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
```

Create an HTML File that Displays "Hello World"

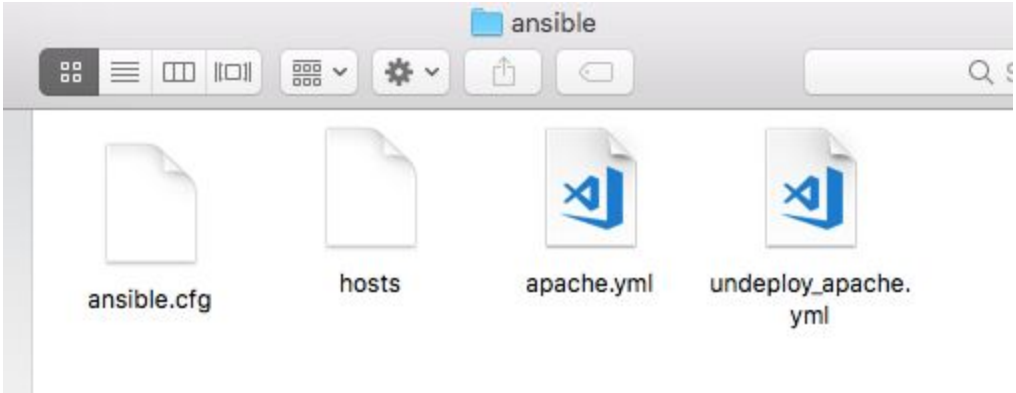
1. On local computer, create a HTML file as following:

```
index.html
1  <!DOCTYPE html>
2  <html>
3  <body>
4
5  <h1>Hello World!</h1>
6
7  </body>
8  </html>
```

Create a Playbook that Deploys a Web Server

1. We created a yaml file in our ansible directory(/etc/ansible) by running the following command:

Sudo nano apache.yml



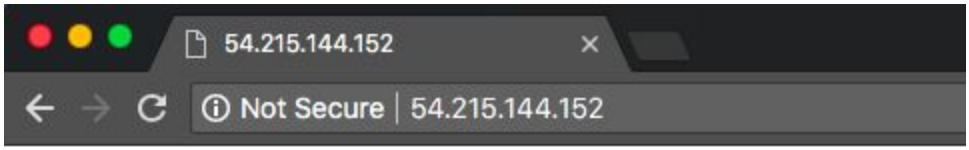
2. We inserted the following code into the file:

```
apache.yml
1   ---
2   - hosts: ec2
3     sudo:  yes
4     tasks:
5         - name: Install apache2
6           raw: sudo apt-get -y install apache2
7         - name: Start Apache
8           raw: sudo service apache2 restart
9         - name: run Hello World webpage
10          copy: src=/Users/ziyun/Desktop/index.html dest=/var/www/html/index.html
11
```

Deploy playbook

1. Use following command to deploy:
   ansible-playbook apache.yml
2. After that, you should be able to open your web using IP address in browser.

Here is a screenshot of our webpage that running on our web server:



Hello World!

Create a Playbook to Un-deploys a Web Server

1. Again create a yaml file in our ansible directory by running the following command:

   Sudo nano undeploy_apache.yml
2. Insert the following content to undeploy apache:

undeploy_apache.yml

```
---
- hosts: ec2
  sudo:  yes
  tasks:
      - name: uninstall apache2
        raw: sudo apt-get -y purge apache2 apache2-utils apache2.2-bin
      - name: autoremove
        raw: sudo apt-get -y autoremove
```

Un-deploy playbook

3. Use following command to deploy:

   ansible-playbook undeploy_apache.yml
4. After that, you should be able to disconnect your web.

Here is a screenshot after executing the uninstall yaml file: