Build an EC2 using Ansible Step By Step

This article explains step by step how to create (spin up) an EC2 instance within AWS using Ansible and a few extras. Unlike 100% of other articles out there, this one actually demonstrates how to do it. Pay attention to the date of this article because things DO change over time.

We’re using CentOS 7 as the Ansible host.

Run the following commands to install the required dependencies for Ansible and AWS. I’m running this as a normal user with SUDO access, not the root user.

sudo pip install --upgrade pip

sudo pip install boto

sudo yum install ansible

Log into your AWS account to get your “AWS\_ACCESS\_KEY\_ID” and “AWS\_SECRET\_ACCESS\_KEY”. Go to “Identity and Access Management”. Create a new user or select an exiting one. Go to “Security Credentials” and click “Create Access Key”. Here’s an example of what you’ll end up with:

Access Key ID: NUHKOIJFOJF9GFJDO

Secret Access Key: LSDJKFODSJF9SDJF8UH3U3HFKW

Keep those safe – download when asked. Use the above values to create environment variables. Copy and paste the following (with your values replacing mine) into your shell:

export AWS\_ACCESS\_KEY\_ID="NUHKOIJFOJF9GFJDO"

export AWS\_SECRET\_ACCESS\_KEY="LSDJKFODSJF9SDJF8UH3U3HFKW"

Create the “~/hosts” file with the following contents:

[local]

localhost

[webserver]

Now we build our YML file for Ansible to run through. Here’s a sample that will create a basic EC2 with a public IP address and your public SSH key. Put the following into the file “~/ec2-basic.yml”

---

- name: Provision an EC2 Instance

hosts: local

connection: local

gather\_facts: False

tags: provisioning

# Necessary Variables for creating/provisioning the EC2 Instance

vars:

instance\_type: t2.micro

security\_group: ansible-webserver # Change the security group name here

image: ami-719fb712 # This is an AMI i created myself

keypair: agix-key # This is one of my keys that i already have in AWS

region: ap-southeast-2 # Change the Region

count: 1

# Task that will be used to Launch/Create an EC2 Instance

tasks:

- name: Create a security group

local\_action:

module: ec2\_group

name: "{{ security\_group }}"

description: Security Group for webserver Servers

region: "{{ region }}"

rules:

- proto: tcp

from\_port: 22

to\_port: 22

cidr\_ip: 0.0.0.0/0

- proto: tcp

from\_port: 80

to\_port: 80

cidr\_ip: 0.0.0.0/0

- proto: tcp

from\_port: 443

to\_port: 443

cidr\_ip: 0.0.0.0/0

rules\_egress:

- proto: all

cidr\_ip: 0.0.0.0/0

register: basic\_firewall

- name: Launch the new EC2 Instance

local\_action: ec2

group={{ security\_group }}

instance\_type={{ instance\_type}}

image={{ image }}

wait=true

region={{ region }}

keypair={{ keypair }}

count={{count}}

register: ec2

- name: Add the newly created EC2 instance(s) to the local host group (located inside the directory)

local\_action: lineinfile

dest="./hosts"

regexp={{ item.public\_ip }}

insertafter="[webserver]" line={{ item.public\_ip }}

with\_items: ec2.instances

- name: Wait for SSH to come up

local\_action: wait\_for

host={{ item.public\_ip }}

port=22

state=started

with\_items: ec2.instances

- name: Add tag to Instance(s)

local\_action: ec2\_tag resource={{ item.id }} region={{ region }} state=present

with\_items: ec2.instances

args:

tags:

Name: webserver

Being the provisioning (spin it up):

ansible-playbook -i ./hosts ec2-basic.yml

And finally log into your new ec2 instance:

ssh -l centos 54.1.2.3 -i Downloads/agix-key.pem