

ANSIBLE

Ansible is a configuration management tool

Ansible:

Log into the Amazon AWS console and launch an Ubuntu instance.

https://us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#Instances:sort=instanceId

Services Resource Groups

Launch Instance Connect Actions

Filter by tags and attributes or search by keyword

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
	i-0cb98c429e98ca7ee	t2.micro	us-east-2b	running	2/2 checks ...	None	ec2-52-15-113-196.us-

Instance: i-0cb98c429e98ca7ee Public DNS: ec2-52-15-113-196.us-east-2.compute.amazonaws.com

Description Status Checks Monitoring Tags

Instance ID	i-0cb98c429e98ca7ee	Public DNS (IPv4)	ec2-52-15-113-196.us-east-2.compute.amazonaws.com
Instance state	running	IPv4 Public IP	52.15.113.196
Instance type	t2.micro	IPv6 IPs	-
Elastic IPs		Private DNS	ip-172-31-25-134.us-east-2.compute.internal

SSH into the instance using command prompt and install Ansible. Follow the commands in the Ansible documentation as outlined below.

To configure the PPA on your machine and install ansible run these commands:

```
$ sudo apt update
$ sudo apt install software-properties-common
$ sudo apt-add-repository --yes --update ppa:ansible/ansible
$ sudo apt install ansible
```

Note

docs.ansible.com/ansible/latest/installation_guide/intro_installation.html#latest-releases-via-apt-ubuntu

```
Select root@ip-172-31-25-134: ~
Microsoft Windows [Version 10.0.18362.356]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\mohan>cd Downloads

C:\Users\mohan\Downloads>ssh -i Ansible.pem ubuntu@ec2-52-15-113-196.us-east-2.compute.amazonaws.com
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
WARNING: UNPROTECTED PRIVATE KEY FILE!                                          @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
Permissions for 'Ansible.pem' are too open.
It is required that your private key files are NOT accessible by others.
This private key will be ignored.
Load key "Ansible.pem": bad permissions
ubuntu@ec2-52-15-113-196.us-east-2.compute.amazonaws.com: Permission denied (publickey).
```

```
ch: Select root@ip-172-31-25-134: ~
Memory usage: 14%          IP address for eth0: 172.31.25.134
Swap usage: 0%

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

Last login: Tue Sep 24 10:01:36 2019 from 96.225.59.182
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-25-134:~$ sudo su -
root@ip-172-31-25-134:~# apt update
Hit:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu bionic InRelease
Get:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Get:4 http://us-east-2.ec2.archive.ubuntu.com/ubuntu bionic/universe amd64 Packages [8570 kB]
Get:5 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Get:6 http://us-east-2.ec2.archive.ubuntu.com/ubuntu bionic/universe Translation-en [4941 kB]
Get:7 http://us-east-2.ec2.archive.ubuntu.com/ubuntu bionic/multiverse amd64 Packages [151 kB]
Get:8 http://us-east-2.ec2.archive.ubuntu.com/ubuntu bionic/multiverse Translation-en [108 kB]
```

Ansible installation completed

```
ch: Select root@ip-172-31-25-134: ~
etting up libpython-stdlib:amd64 (2.7.15~rc1-1) ...
etting up python (2.7.15~rc1-1) ...
etting up python-idna (2.6-1) ...
etting up python-yaml (3.12-1build2) ...
etting up python-asn1crypto (0.24.0-1) ...
etting up python-crypto (2.6.1-8ubuntu2) ...
etting up python-pyasn1 (0.4.2-3) ...
etting up python-pkg-resources (39.0.1-2) ...
etting up python-markupsafe (1.0-1build1) ...
etting up python-httplib2 (0.9.2+dfsg-1ubuntu0.1) ...
etting up python-cffi-backend (1.11.5-1) ...
etting up python-six (1.11.0-2) ...
etting up python-enum34 (1.1.6-2) ...
etting up python-ipaddress (1.0.17-1) ...
etting up python-setuptools (39.0.1-2) ...
etting up python-jinja2 (2.10-1ubuntu0.18.04.1) ...
etting up python-cryptography (2.1.4-1ubuntu1.3) ...
etting up python-paramiko (2.0.0-1ubuntu1.2) ...
etting up ansible (2.8.5-1ppa~bionic) ...
oot@ip-172-31-25-134:~#
```

Create a user named ansible along with the password

```
Select root@ip-172-31-25-134: ~
Setting up libpython-stdlib:amd64 (2.7.15~rc1-1) ...
Setting up python (2.7.15~rc1-1) ...
Setting up python-idna (2.6-1) ...
Setting up python-yaml (3.12-1build2) ...
Setting up python-asn1crypto (0.24.0-1) ...
Setting up python-crypto (2.6.1-8ubuntu2) ...
Setting up python-pyasn1 (0.4.2-3) ...
Setting up python-pkg-resources (39.0.1-2) ...
Setting up python-markupsafe (1.0-1build1) ...
Setting up python-httplib2 (0.9.2+dfsg-1ubuntu0.1) ...
Setting up python-cffi-backend (1.11.5-1) ...
Setting up python-six (1.11.0-2) ...
Setting up python-enum34 (1.1.6-2) ...
Setting up python-ipaddress (1.0.17-1) ...
Setting up python-setuptools (39.0.1-2) ...
Setting up python-jinja2 (2.10-1ubuntu0.18.04.1) ...
Setting up python-cryptography (2.1.4-1ubuntu1.3) ...
Setting up python-paramiko (2.0.0-1ubuntu1.2) ...
Setting up ansible (2.8.5-1ppa~bionic) ...
root@ip-172-31-25-134:~# su - ansible
No passwd entry for user 'ansible'
root@ip-172-31-25-134:~# useradd -m -s /bin/bash ansible
root@ip-172-31-25-134:~# passwd ansible
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
root@ip-172-31-25-134:~#
```

It is advisable not to work as the root.

```
passwd: password updated successfully
root@ip-172-31-25-134:~# whoami
root
root@ip-172-31-25-134:~#
```

Switch to user Ansible in order to perform software installations etc.

Ansible server uses ssh to connect to a remote machine hence create private/public key and public key import into aws account.

Create key pair inside the Ansible key and private key should be kept absolutely secure.

```

ansible@ip-172-31-25-134: ~
Retype new UNIX password:
passwd: password updated successfully
root@ip-172-31-25-134:~# whoami
root
root@ip-172-31-25-134:~# su - ansible
ansible@ip-172-31-25-134:~$
ansible@ip-172-31-25-134:~$
ansible@ip-172-31-25-134:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ansible/.ssh/id_rsa):
Created directory '/home/ansible/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ansible/.ssh/id_rsa.
Your public key has been saved in /home/ansible/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:16zWJkbflJQkdGhUKGdoa1S1RG3oZfPn7dyRE5X55gQ ansible@ip-172-31-25-134
The key's randomart image is:
+---[RSA 2048]-----+
|      +**B+  o|
|      =.+=ooEo.|
|      o = .++ +o|
|      o. o.. .*|
|      .S + o ==|
|      . = o ++|
|      = + . o+|
|      o o    +|
+---[SHA256]-----+
ansible@ip-172-31-25-134:~$

```

Launch the public key and import it into organization's aws account so that this key is used to launch any machines through the company's AWS account.

```

ansible@ip-172-31-25-134: ~
Your public key has been saved in /home/ansible/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:16zWJkbflJQkdGhUKGdoa1S1RG3oZfPn7dyRE5X55gQ ansible@ip-172-31-25-134
The key's randomart image is:
+---[RSA 2048]-----+
|      +**B+  o|
|      =.+=ooEo.|
|      o = .++ +o|
|      o. o.. .*|
|      .S + o ==|
|      . = o ++|
|      = + . o+|
|      o o    +|
+---[SHA256]-----+
ansible@ip-172-31-25-134:~$ cat /home/ansible/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDvPj5xmbKnuw5rQZvnbq5DMULbfokh5qhuzWl/xL6KKA6PDQR0xa9JrnCa+XI+De8MQ0yVR1S7Hw+eA8M
TNUZFL27N7bhU4JWALCvYL39PwVFFEqdRSEng1p+pzSnhFvZwsve60gxbQXdjweU8Kite4/4IHAB04zgcprEJCCA0UzNkOVcJmoG0IT1wtzN RAM 11.0
UX5M0/KQc9/wQLlq9G/gRN70jPJAnS52KY8Ept/B8Fok9R3Pygwi5cjVRTH+BxswvYDuIM6oJO+3p5F1B/81Wj1g2/ZrZgCt+18qn9IMoI+Tf
XuK9M58RTG5saEigpbwv77R ansible@ip-172-31-25-134
ansible@ip-172-31-25-134:~$

```

Copy the following keypair and import key into AWS account. Next time this key will be used to launch instances.

Import Key Pair



Click Browse and navigate to your public key. You may change the name of your key if necessary. Alternatively, you can copy and paste the contents of your public key into the dialog.

Load public key from file

Browse...

Key pair name

Ans

Public key contents

```
AAAAB3NzaC1yc2EAAAADAQABAAQDvPj5xmbKnuw5rQZvnbq5DMULbfohk5qhuzWh/  
xL6KKA6PDQR0xa9JrnCa+XI+De8MQ0yVR1S7Hw+eA8MTNUZFL27N7bhU4JWALCvYL39  
PwvVFFEQdRSEng1p+pzSnvHfVZwsve60gxbQXdjweU8Kite4/4IHabo4zgcpREJCCAOUzNkO  
VcJmoG0IT1WtzN351twGPeYUX5M0/KQc9/wQLlq9G/gRN70jPJaNS52KY8Ept/B8Fok9R3Py  
gwi5cjVRTH+BxsWvvYDuIM6oJO+3p5F1B/8lWj1g2/ZrZgCt+18qn9IMol+TfGICQ0FQ5sXuK9  
M58RTG5saEigpbwv77R ansible@ip-172-31-25-134
```

Cancel

Import

Launch two AWS ubuntu instances using the Ans Key Pair created and the Ansible server will have access to those instances.

Filter by tags and attributes or search by keyword								
	Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
<input checked="" type="checkbox"/>		i-01cd7fa1a8d431c88	t2.micro	us-east-2b	running	Initializing	None	ec2-18-217-218-102.us-east-2.amazonaws.com
<input type="checkbox"/>		i-0cb98c429e98ca7ee	t2.micro	us-east-2b	running	2/2 checks ...	None	ec2-52-15-113-196.us-east-2.amazonaws.com
<input type="checkbox"/>		i-0e8b4b03552807a2c	t2.micro	us-east-2b	running	Initializing	None	ec2-3-16-55-142.us-east-2.amazonaws.com

Next step is to ssh into the two newly created ubuntu instances as ansible user. SSh user@ip address command should be used to ssh between two linux machines. Use the private ip to loginto the machine. There is no need to specify the command ssh -i /home/ansible/.ssh/id_rsa

```
NUZFL27N/bhU4JWALCvYL39PwVVFEEqdRSEng1p+pzSnhVfVZwsve60gxbQXdjweU8K1te4/41HaB04zgcpREJCCA0UznK0VCJmoG01T1WtZN351EwGPey
X5M0/KQc9/wQLlq9G/gRN70jPJaNS52KY8Ept/B8Fok9R3Pygwi5cjVRTH+BxsWvYDuIM6oJO+3p5F1B/8lWj1g2/ZrZgCt+18qn9IMoI+TfGICQ0FQ5s
uK9M58RTG5saEigpbwv77R ansible@ip-172-31-25-134
ansible@ip-172-31-25-134:~$ ssh ubuntu@ec2-18-217-218-102.us-east-2.compute.amazonaws.com
The authenticity of host 'ec2-18-217-218-102.us-east-2.compute.amazonaws.com (172.31.29.101)' can't be established.
ECDSA key fingerprint is SHA256:w3QnxJOZbNQq+OVv+wG3B605xPmQyW7cYNETyldCh5I.
Are you sure you want to continue connecting (yes/no)? y
Please type 'yes' or 'no': yes
Warning: Permanently added 'ec2-18-217-218-102.us-east-2.compute.amazonaws.com,172.31.29.101' (ECDSA) to the list of kn
own hosts.
Enter passphrase for key '/home/ansible/.ssh/id_rsa':
Welcome to Ubuntu 18.04.2 LTS (GNU/Linux 4.15.0-1044-aws x86_64)

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

System information as of Tue Sep 24 11:30:46 UTC 2019

System load: 0.0          Processes:           85
Usage of /:   13.6% of 7.69GB    Users logged in:   0
```

Log into the private ips of both the instances one after the other. Up until now Ansible server was able to ssh into two instances using the public/private keypair. Next step is for Ansible server to ssh into existing instance which happens to be an ansible server itself.

```
ubuntu@ip-172-31-25-134: ~
ubuntu@ip-172-31-25-134:~$ su - ansible
Password:
ansible@ip-172-31-25-134:~$ ssh ubuntu@172.31.25.134
Enter passphrase for key '/home/ansible/.ssh/id_rsa':
Welcome to Ubuntu 18.04.2 LTS (GNU/Linux 4.15.0-1044-aws x86_64)

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

System information as of Wed Sep 25 07:43:54 UTC 2019

System load: 0.0          Processes:           95
Usage of /:   21.7% of 7.69GB    Users logged in:   1
Memory usage: 20%          IP address for eth0: 172.31.25.134
Swap usage:   0%

 * Congrats to the Kubernetes community on 1.16 beta 1! Now available
   in MicroK8s for evaluation and testing, with upgrades to RC and GA

snap info microk8s
```

```
Select ubuntu@ip-172-31-18-85: ~
ubuntu@ip-172-31-18-85:~$ logout
Connection to ec2-3-16-55-142.us-east-2.compute.amazonaws.com closed.
ansible@ip-172-31-25-134:~$ ssh ubuntu@ip-172-31-18-85.us-east-2.compute.internal
The authenticity of host 'ip-172-31-18-85.us-east-2.compute.internal (172.31.18.85)' can't be
ECDSA key fingerprint is SHA256:t19WnbqZm02ZBQhCDc9czFVclLxg05ZJ8iP/Slw14TE.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ip-172-31-18-85.us-east-2.compute.internal' (ECDSA) to the list of known hosts.
Enter passphrase for key '/home/ansible/.ssh/id_rsa':
Welcome to Ubuntu 18.04.2 LTS (GNU/Linux 4.15.0-1044-aws x86_64)

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

System information as of Tue Sep 24 12:59:17 UTC 2019

System load:  0.0          Processes:      87
Usage of /:   13.7% of 7.69GB Users logged in:  0
Memory usage: 14%         IP address for eth0: 172.31.18.85
Swap usage:   0%
```

Copy the public key. Under the Ubuntu user.

Logout and login as Ubuntu user.

```
ansible@ip-172-31-25-134: ~
in MicroK8s for evaluation and testing, with upgrades to RC and GA

snap info microk8s

packages can be updated.
updates are security updates.

Last login: Tue Sep 24 12:57:21 2019 from 172.31.25.134
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-18-85:~$
ubuntu@ip-172-31-18-85:~$ logout
Connection to ip-172-31-18-85.us-east-2.compute.internal closed.
ansible@ip-172-31-25-134:~$ cat .ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDvPj5xmbKnuw5rQZvnbq5DMULbfohk5qhuzWh/xL6KKA6PDQR0xa9JrnCa+XI+De8MQ0yVR1S7Hw+eA8M
NUZFL27N7bhU4JWALCvYL39PwVFFEqdRSEng1p+pzSnvHfVZwsve60gxbQXdjweU8Kite4/4IHAB04zgcpREJCCA0UzNkOVcJmoG0IT1Wtzn351twGPey
X5M0/KQc9/wQLlq9G/gRN70jPJaNS52KY8Ept/B8Fok9R3Pygwi5cjVRTH+BxswvVYDuIM6oJO+3p5F1B/81Wj1g2/ZrZgCt+18qn9IMoI+TfGICQ0FQ5s
uK9M58RTG5saEigpbwv77R ansible@ip-172-31-25-134
ansible@ip-172-31-25-134:~$
```

Copy and paste the key in the `authorized_keys` section.

Run the command `ansible` when logged in as an `ansible` user. A folder named `ansible` would have been created.

```
ansible@ip-172-31-25-134:~$ logout
root@ip-172-31-25-134:~# logout
ubuntu@ip-172-31-25-134:~$ nano .ssh/authorized_keys
ubuntu@ip-172-31-25-134:~$ sudo su - ansible
ansible@ip-172-31-25-134:~$
```



```

Select ansible@ip-172-31-25-134: ~
*** System restart required ***
Last login: Wed Sep 25 02:12:15 2019 from 96.225.59.182
ubuntu@ip-172-31-25-134:~$
ubuntu@ip-172-31-25-134:~$
ubuntu@ip-172-31-25-134:~$ su - ansible
Password:
ansible@ip-172-31-25-134:~$ ansible
Usage: ansible <host-pattern> [options]

Define and run a single task 'playbook' against a set of hosts

Options:
  -a MODULE_ARGS, --args=MODULE_ARGS      module arguments
  --ask-vault-pass                          ask for vault password
  -B SECONDS, --background=SECONDS         run asynchronously, failing after X seconds
                                          (default=N/A)
  -C, --check                              don't make any changes; instead, try to predict some
                                          of the changes that may occur
  -D, --diff                              when changing (small) files and templates, show the

ansible@ip-172-31-25-134:~$ cd .ansible
ansible@ip-172-31-25-134:~/ansible$ pwd
/home/ansible/.ansible
ansible@ip-172-31-25-134:~/ansible$
tmp
ansible@ip-172-31-25-134:~/ansible$ cd /etc/ansible
ansible@ip-172-31-25-134:/etc/ansible$ ls
ansible.cfg  hosts  roles

```

Now by default the ansible.cfg folder is inside the /etc/ansible folder hence copy the file into the /home/ansible/.ansible

```

ansible@ip-172-31-25-134: ~/ansible
--scp-extra-args=SCP_EXTRA_ARGS
    specify extra arguments to pass to scp only (e.g. -l)
--ssh-extra-args=SSH_EXTRA_ARGS
    specify extra arguments to pass to ssh only (e.g. -R)

Some modules do not make sense in Ad-Hoc (include, meta, etc)
ERROR! Missing target hosts
ansible@ip-172-31-25-134:~$ cd .ansible
ansible@ip-172-31-25-134:~/ansible$ ls
tmp
ansible@ip-172-31-25-134:~/ansible$ cd tmp
ansible@ip-172-31-25-134:~/ansible/tmp$ cd ..
ansible@ip-172-31-25-134:~/ansible$ cd /etc/ansible
ansible@ip-172-31-25-134:/etc/ansible$ ls
ansible.cfg  hosts  roles
ansible@ip-172-31-25-134:/etc/ansible$ cp ansible.cfg /home/ansible/.ansible
ansible@ip-172-31-25-134:/etc/ansible$ cd /home/ansible/.ansible
ansible@ip-172-31-25-134:~/ansible$ ls
ansible.cfg  tmp
ansible@ip-172-31-25-134:~/ansible$

```

Next step, edit the ansible.config file in order to disable the hostkey checking(enables the ansible server to connect to the remote machine without showing any warning).

```
ansible@ip-172-31-25-134: ~/.ansible
GNU nano 2.9.3 ansible.cfg Modified
# inject_facts_as_vars = True

# additional paths to search for roles in, colon separated
roles_path = /etc/ansible/roles

# uncomment this to disable SSH key host checking
host_key_checking = False

# change the default callback, you can only have one 'stdout' type enabled at a time.
stdout_callback = skippy

# Ansible ships with some plugins that require whitelisting,
# this is done to avoid running all of a type by default.
# These setting lists those that you want enabled for your system.
# Custom plugins should not need this unless plugin author specifies it.
```

Also change the location of the files that contains the details of the machine in the inventory line as shown in the screenshot below.

```
ansible@ip-172-31-25-134: ~/.ansible
GNU nano 2.9.3 ansible.cfg Modified
# config file for ansible -- https://ansible.com/
# =====

# nearly all parameters can be overridden in ansible-playbook
# or with command line flags. ansible will read ANSIBLE_CONFIG,
# ansible.cfg in the current working directory, .ansible.cfg in
# the home directory or /etc/ansible/ansible.cfg, whichever it
# finds first

[defaults]

# some basic default values...

inventory = /home/ansible/.ansible/hosts
library = /usr/share/my_modules/
module_utils = /usr/share/my_module_utils/
```

Next create a file called host and enter the details of the web, local and db private ip addresses inside the file. Enter the private address of the instances that the ansible server needs to work with.

ansible@ip-172-31-25-134: ~/.ansible

GNU nano 2.9.3

hosts

[local]

172.31.25.134 name=as

[webservers]

172.31.29.101 name=web001

172.31.25.134

172.31.18.85

[dbservers]

172.31.18.85 name=db001