

---

# **SDS on Amazon EC2 Documentation**

*Release 0.1*

**Hyungro Lee**

**Oct 02, 2017**



---

# Contents

---

<b>1</b>	<b>Get Started</b>	<b>3</b>
1.1	Background . . . . .	3
1.2	AWS Credentials . . . . .	3
1.3	Dynamic Inventory . . . . .	4
1.4	Launching EC2 Instances via Ansible <code>ec2</code> module . . . . .	7
1.5	EC2 Tags for Inventory Groups . . . . .	7
1.6	Converting Inventory Groups . . . . .	13
1.7	Terminating EC2 Instances . . . . .	14
<b>2</b>	<b>Basic Issues</b>	<b>19</b>
<b>3</b>	<b>VPC &amp; Subnet</b>	<b>21</b>
<b>4</b>	<b>Additional Files e.g. <code>ec2.ini</code> or <code>ec2.py</code> for dynamic inventory</b>	<b>23</b>
<b>5</b>	<b>Performance Measurement</b>	<b>25</b>
<b>6</b>	<b><code>ec2.py</code></b>	<b>29</b>
<b>7</b>	<b><code>ec2 facts</code></b>	<b>33</b>
7.1	In Practice . . . . .	36
<b>8</b>	<b>Facts for Create and Terminate Instances</b>	<b>41</b>
8.1	<code>to_nice_yaml</code> . . . . .	41
8.2	<code>include_vars</code> . . . . .	41
<b>9</b>	<b>Indices and tables</b>	<b>43</b>



EC2 instances are created or terminated with the Ansible Plays in this repository, therefore software stack deployment is achieved on Amazon Cloud with dynamic computing resources. Ansible dynamic inventory and cloud modules are used to manage cloud resources such as security groups, VPC (Virtual Private Cloud) with public ip addresses.

Contents:



In this document, we expect to start ec2 instances, deploy software stacks with a dynamic inventory like a text inventory file and terminate instances when its completed. This can be applied to other cloud platforms such as Microsoft Azure or OpenStack.

## Background

Typical ansible playbook runs with a static text inventory file which consists of a set of IP addresses or host names accessible via SSH because it assume that there are machines ready to install software and configure settings through Ansible roles, plays. When it comes to cloud resources, those IPs or host names are not permanent because new instances are started with different network access information. Ansible Dynamic Inventory allows to retrieve instance information directly from EC2 instead of reading a text file.

In addition, Ansible `ec2` cloud module supports creating/terminating ec2 instances as Ansible tasks, therefore booting and terminating ec2 instances are automated by Ansible plays, in this case, `boot.yml` and `terminate.yml` files.

## AWS Credentials

`AWS_ACCESS_KEY` and `AWS_SECRET_ACCESS_KEY` are required to communicate with AWS under your account. Ansible provides several options to import credentials like `ec2.ini` file, boto profile (e.g. `~/aws/credentials`) and environment variables but we will use environment variables with `ansible-vault` to secure the information.

- Be ready with your `AWS_ACCESS_KEY` and `AWS_SECRET_ACCESS_KEY`
- Store the information in a file like:

```
$ nano cred
export AWS_ACCESS_KEY=1234567890
export AWS_SECRET_ACCESS_KEY=1234567890
```

(Replace 1234567890 with a real value)

- Encrypt the file with `ansible-vault` with your password for this file:

```
$ ansible-vault encrypt cred
New Vault password:
Confirm New Vault password:
```

The encrypted file looks like:

```
$ cat cred
$ANSIBLE_VAULT;1.1;AES256
36383835346339373263616238633662613265653766646236616538326130616666373864323537
6265613637646530396663393236636534313138623536300a343638643134656330646661653230
32363234646631323034336338376337323836616532396639656234396235303531306133323663
3838383962386338320a623932313934356563303964303831323732323363343364646235633261
61396361366131363262623739376535393062663361303966333533396664663832616234306234
34623461316530643063313338666635396463336361643962633536666264613665343135303163
613937373831323931613039653438363431
```

- Enable your EC2 credentials in a shell by:

```
$ $(ansible-vault view cred)
Vault password:
$
```

You can confirm whether it's loaded or not by:

```
$ env|grep AWS
AWS_SECRET_ACCESS_KEY=1234567890
AWS_ACCESS_KEY=1234567890
```

## Dynamic Inventory

It is not required to have a text inventory file in your Ansible home directory as long as `ec2.py` script provides dynamic inventory.

- Download and run `ec2.py` AWS External Inventory Script:

```
$ curl -L https://raw.githubusercontent.com/ansible/ansible/devel/contrib/inventory/ec2.py > ec2.py
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
Dload  Upload  Total    Spent    Left  Speed
0      0    0     0    0     0      0      0  --:--:--  --:--:--  --:--:--    0
100 63776 100 63776    0     0  135k    0  --:--:--  --:--:--  --:--:--  317k
```

- Make it executable by:

```
$ chmod a+x ./ec2.py
```

- Check running EC2 instances, (wait a few seconds):

```
$ ./ec2.py
{
  "_meta": {
    "hostvars": {}
  }
}
```



There is no running instances now but if you have some, the JSON return dataset looks like:

```
{
  "_meta": {
    "hostvars": {
      "54.85.130.171": {
        "ansible_ssh_host": "54.85.130.171",
        "ec2__in_monitoring_element": false,
        "ec2_ami_launch_index": "0",
        "ec2_architecture": "x86_64",
        "ec2_client_token": "",
        "ec2_dns_name": "",
        "ec2_ebs_optimized": false,
        "ec2_eventsSet": "",
        "ec2_group_name": "",
        "ec2_hypervisor": "xen",
        "ec2_id": "i-1a2c68e2",
        "ec2_image_id": "ami-2d39803a",
        "ec2_instance_profile": "",
        "ec2_instance_type": "t2.micro",
        "ec2_ip_address": "54.85.130.171",
        "ec2_item": "",
        "ec2_kernel": "",
        "ec2_key_name": "albert",
        "ec2_launch_time": "2016-09-15T17:18:36.000Z",
        "ec2_monitored": false,
        "ec2_monitoring": "",
        "ec2_monitoring_state": "disabled",
        "ec2_persistent": false,
        "ec2_placement": "us-east-1a",
        "ec2_platform": "",
        "ec2_previous_state": "",
        "ec2_previous_state_code": 0,
        "ec2_private_dns_name": "ip-172-30-3-104.ec2.internal",
        "ec2_private_ip_address": "172.30.3.104",
        "ec2_public_dns_name": "",
        "ec2_ramdisk": "",
        "ec2_reason": "",
        "ec2_region": "us-east-1",
        "ec2_requester_id": "",
        "ec2_root_device_name": "/dev/sda1",
        "ec2_root_device_type": "ebs",
        "ec2_security_group_ids": "sg-69ebc60c,sg-115b376b",
        "ec2_security_group_names": "default,abds_secgroup",
        "ec2_sourceDestCheck": "true",
        "ec2_spot_instance_request_id": "",
        "ec2_state": "running",
        "ec2_state_code": 16,
        "ec2_state_reason": "",
        "ec2_subnet_id": "subnet-719a774d",
        "ec2_tag_datanodes": "True",
        "ec2_tag_frontendnodes": "False",
        "ec2_tag_hadoopnodes": "True",
        "ec2_tag_historyservernodes": "False",
        "ec2_tag_journalnodes": "True",
        "ec2_tag_namenodes": "True",
        "ec2_tag_resourcemanagernodes": "True",
        "ec2_tag_zookeepernodes": "True",
        "ec2_virtualization_type": "hvm",
```

```
        "ec2_vpc_id": "vpc-e6c17c83"
    }
}
},
"ami_2d39803a": [
    "54.85.130.171"
],
"ec2": [
    "54.85.130.171"
],
"i-1a2c68e2": [
    "54.85.130.171"
],
"key_albert": [
    "54.85.130.171"
],
"security_group_abds_secgroup": [
    "54.85.130.171"
],
"security_group_default": [
    "54.85.130.171"
],
"tag_datanodes_True": [
    "54.85.130.171"
],
"tag_frontendnodes_False": [
    "54.85.130.171"
],
"tag_hadoopnodes_True": [
    "54.85.130.171"
],
"tag_historyservernodes_False": [
    "54.85.130.171"
],
"tag_journalnodes_True": [
    "54.85.130.171"
],
"tag_namenodes_True": [
    "54.85.130.171"
],
"tag_resourcemanagernodes_True": [
    "54.85.130.171"
],
"tag_zookeepernodes_True": [
    "54.85.130.171"
],
"type_t2_micro": [
    "54.85.130.171"
],
"us-east-1": [
    "54.85.130.171"
],
"us-east-1a": [
    "54.85.130.171"
],
"vpc_id_vpc_e6c17c83": [
    "54.85.130.171"
]
]
```

```
}
```

- Switch inventory argument option in your `ansible-playbook` command like:

```
$ ansible-playbook -i ec2.py
```

For more information, see Ansible Dynamic Inventory [here](#)

## Launching EC2 Instances via Ansible `ec2` module

Ansible provide cloud modules to manage instances and we use `ec2` module for Amazon EC2 cloud to launch or terminate instances. Other clouds i.e. Azure, Digital Ocean, Docker, Google Compute Engine (GCE) or OpenStack are also supported in the cloud modules. For more information see documentation [here](#) and source code from `ansible-modules-core` [here](#)

The following `ec2` task starts a single EC2 instance with Ubuntu 14.04 image:

```
- ec2:
  key_name: albert          # SSH registered key name
  instance_type: t2.micro  # EC2 Instance size
  image: ami-2d39803a     # Ubuntu 14.04 image
  group: default          # security group
  region: us-east-1      # EC2 Region
```

This five lines in `ec2` task launches a new EC2 instance on `us-east-1` region. Find out more details [here](#)

## EC2 Tags for Inventory Groups

`instance_tags` is used to create inventory groups therefore Ansible plays run on particular hosts. For example, deploying hadoop cluster requires several inventory groups such as namenodes, datanodes or zookeepernodes. `instance_tags` is useful to put a label on `ec2` instances to distinguish virtual servers between different inventory groups. The following example shows that three `ec2` instances will be created with the tags for namenodes, datanodes and zookeepernodes like:

```
- ec2:
  key_name: "{{ keypair }}"
  instance_type: "{{ instance_size }}"
  image: "{{ image }}"
  region: "{{ region }}"
  exact_count: 3
  count_tag: three-nodes
  group: [ 'default', "{{ security_group }}" ]
  instance_tags:
    namenodes: True
    datanodes: True
    zookeepernodes: True
    historyservernodes: False
    frontendnodes: False
```

The `key:value` tags are used here and the AWS EC2 returns JSON data like:

```
{u'ansible_all_ipv4_addresses': [u'172.30.3.139'],
 u'ansible_all_ipv6_addresses': [u'fe80::4b9:ffff:fe81:65f'],
```

```
u'ansible_architecture': u'x86_64',
u'ansible_bios_date': u'05/12/2016',
u'ansible_bios_version': u'4.2.amazon',
'ansible_check_mode': False,
u'ansible_cmdline': {u'BOOT_IMAGE': u'/boot/vmlinuz-3.13.0-91-generic',
                    u'console': u'ttys0',
                    u'ro': True,
                    u'root': u'UUID=73362e04-90b8-4c53-b878-2097820e0b34'},
u'ansible_date_time': {u'date': u'2016-09-16',
                      u'day': u'16',
                      u'epoch': u'1473991427',
                      u'hour': u'02',
                      u'iso8601': u'2016-09-16T02:03:47Z',
                      u'iso8601_basic': u'20160916T020347397930',
                      u'iso8601_basic_short': u'20160916T020347',
                      u'iso8601_micro': u'2016-09-16T02:03:47.398008Z',
                      u'minute': u'03',
                      u'month': u'09',
                      u'second': u'47',
                      u'time': u'02:03:47',
                      u'tz': u'UTC',
                      u'tz_offset': u'+0000',
                      u'weekday': u'Friday',
                      u'weekday_number': u'5',
                      u'weeknumber': u'37',
                      u'year': u'2016'},
u'ansible_default_ipv4': {u'address': u'172.30.3.139',
                         u'alias': u'eth0',
                         u'broadcast': u'172.30.3.255',
                         u'gateway': u'172.30.3.1',
                         u'interface': u'eth0',
                         u'macaddress': u'06:b9:ff:81:06:5f',
                         u'mtu': 9001,
                         u'netmask': u'255.255.255.0',
                         u'network': u'172.30.3.0',
                         u'type': u'ether'},
u'ansible_default_ipv6': {},
u'ansible_devices': {u'xvda': {u'holders': [],
                              u'host': u'',
                              u'model': None,
                              u'partitions': {u'xvda1': {u'sectors': u'16755795',
                                                         u'sectorsize': 512,
                                                         u'size': u'7.99 GB',
                                                         u'start': u'16065'}}},
                    u'removable': u'0',
                    u'rotational': u'0',
                    u'sas_address': None,
                    u'sas_device_handle': None,
                    u'scheduler_mode': u'deadline',
                    u'sectors': u'16777216',
                    u'sectorsize': u'512',
                    u'size': u'8.00 GB',
                    u'support_discard': u'0',
                    u'vendor': None}},
u'ansible_distribution': u'Ubuntu',
u'ansible_distribution_major_version': u'14',
u'ansible_distribution_release': u'trusty',
u'ansible_distribution_version': u'14.04',
```

```

u'ansible_dns': {u'nameservers': [u'172.30.0.2'],
  u'search': [u'ec2.internal']},
u'ansible_domain': u'',
u'ansible_env': {u'HOME': u'/home/ubuntu',
  u'LANG': u'en_US.UTF-8',
  u'LC_ALL': u'en_US.UTF-8',
  u'LC_MESSAGES': u'en_US.UTF-8',
  u'LOGNAME': u'ubuntu',
  u'MAIL': u'/var/mail/ubuntu',
  u'PATH': u'/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/
↪bin:/usr/games:/usr/local/games',
  u'PWD': u'/home/ubuntu',
  u'SHELL': u'/bin/bash',
  u'SHLVL': u'1',
  u'SSH_CLIENT': u'156.56.102.202 38016 22',
  u'SSH_CONNECTION': u'156.56.102.202 38016 172.30.3.139 22',
  u'SSH_TTY': u'/dev/pts/0',
  u'TERM': u'xterm',
  u'USER': u'ubuntu',
  u'XDG_RUNTIME_DIR': u'/run/user/1000',
  u'XDG_SESSION_ID': u'8',
  u'_': u'/bin/sh'},
u'ansible_eth0': {u'active': True,
  u'device': u'eth0',
  u'ipv4': {u'address': u'172.30.3.139',
    u'broadcast': u'172.30.3.255',
    u'netmask': u'255.255.255.0',
    u'network': u'172.30.3.0'},
  u'ipv6': [{u'address': u'fe80::4b9:ffff:fe81:65f',
    u'prefix': u'64',
    u'scope': u'link'}],
  u'macaddress': u'06:b9:ff:81:06:5f',
  u'mtu': 9001,
  u'pciid': u'vif-0',
  u'promisc': False,
  u'type': u'ether'},
u'ansible_fips': False,
u'ansible_form_factor': u'Other',
u'ansible_fqdn': u'ip-172-30-3-139',
u'ansible_gather_subset': [u'hardware', u'network', u'virtual'],
u'ansible_hostname': u'ip-172-30-3-139',
u'ansible_interfaces': [u'lo', u'eth0'],
u'ansible_kernel': u'3.13.0-91-generic',
u'ansible_lo': {u'active': True,
  u'device': u'lo',
  u'ipv4': {u'address': u'127.0.0.1',
    u'broadcast': u'host',
    u'netmask': u'255.0.0.0',
    u'network': u'127.0.0.0'},
  u'ipv6': [{u'address': u '::1',
    u'prefix': u'128',
    u'scope': u'host'}],
  u'mtu': 65536,
  u'promisc': False,
  u'type': u'loopback'},
u'ansible_lsb': {u'codename': u'trusty',
  u'description': u'Ubuntu 14.04.4 LTS',
  u'id': u'Ubuntu',

```

```

        u'major_release': u'14',
        u'release': u'14.04'},
u'ansible_machine': u'x86_64',
u'ansible_machine_id': u'dd1bac9cb23792a632bd6ee157719517',
u'ansible_memfree_mb': 715,
u'ansible_memory_mb': {u'nocache': {u'free': 932, u'used': 60},
                        u'real': {u'free': 715, u'total': 992, u'used': 277},
                        u'swap': {u'cached': 0,
                                  u'free': 0,
                                  u'total': 0,
                                  u'used': 0}},
u'ansible_memtotal_mb': 992,
u'ansible_mounts': [{u'device': u'/dev/xvda1',
                     u'fstype': u'ext4',
                     u'mount': u '/',
                     u'options': u'rw,discard',
                     u'size_available': 7029567488,
                     u'size_total': 8309932032,
                     u'uuid': u''}],
u'ansible_nodename': u'ip-172-30-3-139',
u'ansible_os_family': u'Debian',
u'ansible_pkg_mgr': u'apt',
u'ansible_processor': [u'GenuineIntel',
                       u'Intel(R) Xeon(R) CPU E5-2670 v2 @ 2.50GHz'],
u'ansible_processor_cores': 1,
u'ansible_processor_count': 1,
u'ansible_processor_threads_per_core': 1,
u'ansible_processor_vcpus': 1,
u'ansible_product_name': u'HVM domU',
u'ansible_product_serial': u'NA',
u'ansible_product_uuid': u'NA',
u'ansible_product_version': u'4.2.amazon',
u'ansible_python': {u'executable': u'/usr/bin/python',
                    u'has_sslcontext': False,
                    u'type': u'CPython',
                    u'version': {u'major': 2,
                                  u'micro': 6,
                                  u'minor': 7,
                                  u'releaselevel': u'final',
                                  u'serial': 0},
                    u'version_info': [2, 7, 6, u'final', 0]},
u'ansible_python_version': u'2.7.6',
u'ansible_selinux': False,
u'ansible_service_mgr': u'upstart',
u'ansible_ssh_host': u'52.23.213.103',
u'ansible_ssh_host_key_dsa_public': u'AAAAB3NzaC1kc3MAAACBAKosQQsY22MWjQv/
↪bF7V6kewbM5Z7NIPjYgNL/j81bUrmZv/kZSAArfs/
↪ByKjC2gesSZUi3hZ1Lo1LmFZjcu00IUkKosKnPhHF6hNoyApXHNL1FovWULIWAyAbxpsRNkBNJYQ90x0Sk1gf1PC94GJwL4SD31
↪FVRR4ei2UZC04TEsZ4aXfvGzt6OnVRcguiozjeWnglWhikvcdRQHxsZrziiaDdubnomlfn0xMQAAAIEAgKs1pwbh9S/
↪dE5nbSqwLkTiOR6jkcPhP8XGSvjcjXqbm+ymnjJ9W9hM7DiaFm83YjWf+3+LOLMtULf7w91xOC0HLRD5M2WWQ/
↪RxATcLN37jhbAs3XkYqZNCtVWOG6m/wbOiWt3R3SYS5h6Bp4oqHyTYifQsfY+KVSyb29RjZzc=',
u'ansible_ssh_host_key_ecdsa_public': u
↪'AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBcjuapwx2AgG834ti6pGu7gbF2YJmoQ1r6wGVNQrnVHs7
↪f9WGAQgAn89dWyHPE=',
u'ansible_ssh_host_key_ed25519_public': u'AAAAC3NzaC1lZDI1NTE5AAAAIEOzako99Tn5mO/
↪vjk+ctHc9HKmtsdJxTqqNIoo0RI3',
u'ansible_ssh_host_key_rsa_public': u
↪'AAAAB3NzaC1yc2EAAAADAQABAAQAC7elCMAVXdqIuBrO+4Fx5oPggQdI7JRKv176NsoRtwsYPTxp07t4ml1kH6ATsQgkj9wL
↪i8JXJtzAdvZZNn1cH1zCi1WD3bRr3dp+p8JU+OmVBhaElPYqm3W+BDWGC7b4YqHbqTHDNTUsugvLuXr0Qw7fdWMNxb8Va7c3G
↪',

```

```

u'ansible_swapfree_mb': 0,
u'ansible_swaptotal_mb': 0,
u'ansible_system': u'Linux',
u'ansible_system_capabilities': [u''],
u'ansible_system_capabilities_enforced': u'True',
u'ansible_system_vendor': u'Xen',
u'ansible_uptime_seconds': 21477,
u'ansible_user_dir': u'/home/ubuntu',
u'ansible_user_gecos': u'Ubuntu',
u'ansible_user_gid': 1000,
u'ansible_user_id': u'ubuntu',
u'ansible_user_shell': u'/bin/bash',
u'ansible_user_uid': 1000,
u'ansible_userspace_architecture': u'x86_64',
u'ansible_userspace_bits': u'64',
'ansible_version': {'full': '2.1.1.0',
                    'major': 2,
                    'minor': 1,
                    'revision': 1,
                    'string': '2.1.1.0'},
u'ansible_virtualization_role': u'guest',
u'ansible_virtualization_type': u'xen',
u'ec2__in_monitoring_element': False,
u'ec2_ami_launch_index': u'0',
u'ec2_architecture': u'x86_64',
u'ec2_client_token': u'',
u'ec2_dns_name': u'',
u'ec2_ebs_optimized': False,
u'ec2_eventsSet': u'',
u'ec2_group_name': u'',
u'ec2_hypervisor': u'xen',
u'ec2_id': u'i-435014bb',
u'ec2_image_id': u'ami-2d39803a',
u'ec2_instance_profile': u'',
u'ec2_instance_type': u't2.micro',
u'ec2_ip_address': u'52.23.213.103',
u'ec2_item': u'',
u'ec2_kernel': u'',
u'ec2_key_name': u'albert',
u'ec2_launch_time': u'2016-09-15T20:05:34.000Z',
u'ec2_monitored': False,
u'ec2_monitoring': u'',
u'ec2_monitoring_state': u'disabled',
u'ec2_persistent': False,
u'ec2_placement': u'us-east-1a',
u'ec2_platform': u'',
u'ec2_previous_state': u'',
u'ec2_previous_state_code': 0,
u'ec2_private_dns_name': u'ip-172-30-3-139.ec2.internal',
u'ec2_private_ip_address': u'172.30.3.139',
u'ec2_public_dns_name': u'',
u'ec2_ramdisk': u'',
u'ec2_reason': u'',
u'ec2_region': u'us-east-1',
u'ec2_requester_id': u'',
u'ec2_root_device_name': u'/dev/sda1',
u'ec2_root_device_type': u'ebs',
u'ec2_security_group_ids': u'sg-69ebc60c,sg-115b376b',

```

```

u'ec2_security_group_names': u'default,abds_secgroup',
u'ec2_sourceDestCheck': u'true',
u'ec2_spot_instance_request_id': u'',
u'ec2_state': u'running',
u'ec2_state_code': 16,
u'ec2_state_reason': u'',
u'ec2_subnet_id': u'subnet-719a774d',
u'ec2_tag_datanodes': u'True',
u'ec2_tag_frontendnodes': u'True',
u'ec2_tag_hadoopnodes': u'True',
u'ec2_tag_historyservernodes': u'True',
u'ec2_tag_journalnodes': u'True',
u'ec2_tag_namenodes': u'True',
u'ec2_tag_resourcemanagernodes': u'True',
u'ec2_tag_zookeepernodes': u'True',
u'ec2_virtualization_type': u'hvm',
u'ec2_vpc_id': u'vpc-e6c17c83',
'group_names': [u'ami_2d39803a',
                u'ec2',
                u'i-435014bb',
                u'key_albert',
                u'security_group_abds_secgroup',
                u'security_group_default',
                u'tag_datanodes_True',
                u'tag_frontendnodes_True',
                u'tag_hadoopnodes_True',
                u'tag_historyservernodes_True',
                u'tag_journalnodes_True',
                u'tag_namenodes_True',
                u'tag_resourcemanagernodes_True',
                u'tag_zookeepernodes_True',
                u'type_t2_micro',
                u'us-east-1',
                u'us-east-1a',
                u'vpc_id_vpc_e6c17c83'],
'groups': {'all': [u'52.23.213.103', u'54.196.41.145', u'54.209.137.235'],
           u'ami_2d39803a': [u'52.23.213.103',
                             u'54.196.41.145',
                             u'54.209.137.235'],
           u'ec2': [u'52.23.213.103', u'54.196.41.145', u'54.209.137.235'],
           u'i-435014bb': [u'52.23.213.103'],
           u'i-e250141a': [u'54.196.41.145'],
           u'i-e350141b': [u'54.209.137.235'],
           u'key_albert': [u'52.23.213.103',
                           u'54.196.41.145',
                           u'54.209.137.235'],
           u'security_group_abds_secgroup': [u'52.23.213.103',
                                              u'54.196.41.145',
                                              u'54.209.137.235'],
           u'security_group_default': [u'52.23.213.103',
                                       u'54.196.41.145',
                                       u'54.209.137.235'],
           u'tag_datanodes_True': [u'52.23.213.103',
                                    u'54.196.41.145',
                                    u'54.209.137.235'],
           u'tag_frontendnodes_False': [u'54.196.41.145', u'54.209.137.235'],
           u'tag_frontendnodes_True': [u'52.23.213.103'],
           u'tag_hadoopnodes_True': [u'52.23.213.103'],

```



```

        u'54.196.41.145',
        u'54.209.137.235'],
    u'tag_historyservernodes_False': [u'54.196.41.145',
        u'54.209.137.235'],
    u'tag_historyservernodes_True': [u'52.23.213.103'],
    u'tag_journalnodes_True': [u'52.23.213.103',
        u'54.196.41.145',
        u'54.209.137.235'],
    u'tag_namenodes_True': [u'52.23.213.103',
        u'54.196.41.145',
        u'54.209.137.235'],
    u'tag_resourcemanagernodes_True': [u'52.23.213.103',
        u'54.196.41.145',
        u'54.209.137.235'],
    u'tag_zookeepernodes_True': [u'52.23.213.103',
        u'54.196.41.145',
        u'54.209.137.235'],
    u'type_t2_micro': [u'52.23.213.103',
        u'54.196.41.145',
        u'54.209.137.235'],
    'ungrouped': ['localhost'],
    u'us-east-1': [u'52.23.213.103',
        u'54.196.41.145',
        u'54.209.137.235'],
    u'us-east-1a': [u'52.23.213.103',
        u'54.196.41.145',
        u'54.209.137.235'],
    u'vpc_id_vpc_e6c17c83': [u'52.23.213.103',
        u'54.196.41.145',
        u'54.209.137.235']},
    'inventory_dir': '/home/albert/git/aws-cloudformation-by-ansible',
    'inventory_file': 'ec2.py',
    'inventory_hostname': u'52.23.213.103',
    'inventory_hostname_short': u'52',
    'module_setup': True,
    'omit': '__omit_place_holder__943b1cb4d91b305bccdb59adcbdfd6146fda5e7e',
    'playbook_dir': u'/home/albert/git/aws-cloudformation-by-ansible'}

```

As you noticed that tags create separate host groups in a `tag_<key>_<value>` form, for example, `namenodes` are listed like:

```

u'tag_namenodes_True': [u'52.23.213.103',
    u'54.196.41.145',
    u'54.209.137.235'],

```

It is good because Ansible directly call this group like:

```
$ ansible -m ping -i ec2.py -u ubuntu "tag_namenodes_True"
```

However, if hostnames should be just `namenodes` without any prefix or postfix, we can fix that using `add_host` module. Let's see that in `convert.yml`.

## Converting Inventory Groups

`convert.yml` uses `add_host` to create a new inventory groups and it corrects group names from `tag_<key>_<value>` format to `<key>` like just `namenodes`. Sample task looks like

```

- name: convert EC2 tags_*_True inventory groups
  hosts: "tag_namenodes_True"
  tasks:
    - name: Add new instance to host group (namenodes)
      add_host: hostname="{{ hostvars[item]['ec2_ip_address'] }}"
↳groupname=namenodes
      with_items: "{{ groups['tag_namenodes_True'] }}"

```

This task runs towards `tag_namenodes_True` group and create a new group named `namenodes` with EC2 IP addresses therefore following ansible playbooks or roles can be used without modification in hostnames.

## Terminating EC2 Instances

`state: 'absent'` terminates ec2 instances with instance ids which `boot.yml` stores using `ec2_facts`. Therefore, `terminate.yml` reads the YAML fact file and gets instance ids to terminate. `include_vars` imports data from YAML and makes them accessible in Ansible Plays.

Sample `ec2_fact` file looks like:

```

"ansible_facts": {
  "changed": true,
  "instance_ids": [
    "i-8c064074",
    "i-8306407b"
  ],
  "instances": [
    {
      "ami_launch_index": "0",
      "architecture": "x86_64",
      "block_device_mapping": {
        "/dev/sda1": {
          "delete_on_termination": true,
          "status": "attached",
          "volume_id": "vol-71970cf0"
        }
      },
      "dns_name": "",
      "ebs_optimized": false,
      "groups": {
        "sg-115b376b": "abds_secgroup",
        "sg-69ebc60c": "default"
      },
      "hypervisor": "xen",
      "id": "i-8c064074",
      "image_id": "ami-2d39803a",
      "instance_type": "t2.micro",
      "kernel": null,
      "key_name": "albert",
      "launch_time": "2016-09-17T03:14:11.000Z",
      "placement": "us-east-1a",
      "private_dns_name": "ip-172-30-3-89.ec2.internal",
      "private_ip": "172.30.3.89",
      "public_dns_name": "",
      "public_ip": "54.159.24.94",
      "ramdisk": null,
      "region": "us-east-1",

```

```

    "root_device_name": "/dev/sda1",
    "root_device_type": "ebs",
    "state": "running",
    "state_code": 16,
    "tags": {
      "datanodes": "True",
      "frontendnodes": "False",
      "hadoopnodes": "True",
      "historyservernodes": "False",
      "journalnodes": "True",
      "namenodes": "True",
      "resourcemanagernodes": "True",
      "zookeepernodes": "True"
    },
    "tenancy": "default",
    "virtualization_type": "hvm"
  },
  {
    "ami_launch_index": "1",
    "architecture": "x86_64",
    "block_device_mapping": {
      "/dev/sda1": {
        "delete_on_termination": true,
        "status": "attached",
        "volume_id": "vol-c4960d45"
      }
    },
    "dns_name": "",
    "ebs_optimized": false,
    "groups": {
      "sg-115b376b": "abds_secgroup",
      "sg-69ebc60c": "default"
    },
    "hypervisor": "xen",
    "id": "i-8306407b",
    "image_id": "ami-2d39803a",
    "instance_type": "t2.micro",
    "kernel": null,
    "key_name": "albert",
    "launch_time": "2016-09-17T03:14:11.000Z",
    "placement": "us-east-1a",
    "private_dns_name": "ip-172-30-3-88.ec2.internal",
    "private_ip": "172.30.3.88",
    "public_dns_name": "",
    "public_ip": "54.175.124.32",
    "ramdisk": null,
    "region": "us-east-1",
    "root_device_name": "/dev/sda1",
    "root_device_type": "ebs",
    "state": "running",
    "state_code": 16,
    "tags": {
      "datanodes": "True",
      "frontendnodes": "False",
      "hadoopnodes": "True",
      "historyservernodes": "False",
      "journalnodes": "True",
      "namenodes": "True",

```

```

        "resourcemanagernodes": "True",
        "zookeepernodes": "True"
    },
    "tenancy": "default",
    "virtualization_type": "hvm"
}
],
"tagged_instances": [
    {
        "ami_launch_index": "0",
        "architecture": "x86_64",
        "block_device_mapping": {
            "/dev/sda1": {
                "delete_on_termination": true,
                "status": "attached",
                "volume_id": "vol-71970cf0"
            }
        },
        "dns_name": "",
        "ebs_optimized": false,
        "groups": {
            "sg-115b376b": "abds_secgroup",
            "sg-69ebc60c": "default"
        },
        "hypervisor": "xen",
        "id": "i-8c064074",
        "image_id": "ami-2d39803a",
        "instance_type": "t2.micro",
        "kernel": null,
        "key_name": "albert",
        "launch_time": "2016-09-17T03:14:11.000Z",
        "placement": "us-east-1a",
        "private_dns_name": "ip-172-30-3-89.ec2.internal",
        "private_ip": "172.30.3.89",
        "public_dns_name": "",
        "public_ip": "54.159.24.94",
        "ramdisk": null,
        "region": "us-east-1",
        "root_device_name": "/dev/sda1",
        "root_device_type": "ebs",
        "state": "running",
        "state_code": 16,
        "tags": {
            "datanodes": "True",
            "frontendnodes": "False",
            "hadoopnodes": "True",
            "historyservernodes": "False",
            "journalnodes": "True",
            "namenodes": "True",
            "resourcemanagernodes": "True",
            "zookeepernodes": "True"
        },
        "tenancy": "default",
        "virtualization_type": "hvm"
    },
    {
        "ami_launch_index": "1",
        "architecture": "x86_64",

```

```

    "block_device_mapping": {
      "/dev/sda1": {
        "delete_on_termination": true,
        "status": "attached",
        "volume_id": "vol-c4960d45"
      }
    },
    "dns_name": "",
    "ebs_optimized": false,
    "groups": {
      "sg-115b376b": "abds_secgroup",
      "sg-69ebc60c": "default"
    },
    "hypervisor": "xen",
    "id": "i-8306407b",
    "image_id": "ami-2d39803a",
    "instance_type": "t2.micro",
    "kernel": null,
    "key_name": "albert",
    "launch_time": "2016-09-17T03:14:11.000Z",
    "placement": "us-east-1a",
    "private_dns_name": "ip-172-30-3-88.ec2.internal",
    "private_ip": "172.30.3.88",
    "public_dns_name": "",
    "public_ip": "54.175.124.32",
    "ramdisk": null,
    "region": "us-east-1",
    "root_device_name": "/dev/sda1",
    "root_device_type": "ebs",
    "state": "running",
    "state_code": 16,
    "tags": {
      "datanodes": "True",
      "frontendnodes": "False",
      "hadoopnodes": "True",
      "historyservernodes": "False",
      "journalnodes": "True",
      "namenodes": "True",
      "resourcemanagernodes": "True",
      "zookeepernodes": "True"
    },
    "tenancy": "default",
    "virtualization_type": "hvm"
  }
]
},
"changed": false,
"invocation": {
  "module_args": {
    "file": "/tmp/a.yml"
  },
  "module_name": "include_vars"
}
}

```



## CHAPTER 2

---

### Basic Issues

---

- secgroup is only available for vpc
- vpc or subnetid is necessary to create ec2 instance
- '{{secgroup.name}}' doesn't work even though it is defined from [http://docs.ansible.com/ansible/ec2\\_group\\_module.html](http://docs.ansible.com/ansible/ec2_group_module.html) Did I something wrong?
- group\_id works!:

```
31         group_id: "{{secgroup_first.group_id}}"
```





## CHAPTER 3

---

### VPC & Subnet

---

- manually added from portal website for those



---

### Additional Files e.g. ec2.ini or ec2.py for dynamic inventory

---

aws\_access\_key\_id and aws\_secret\_key need to be defined in the ec2.ini file. additionally, rds and elasticache need to be set False:

```
rds = False  
elasticache = False
```



---

## Performance Measurement

---

Time lapse can be measured by callback functions defined in `ansible.cfg`

```
[defaults]
callback_whitelist = profile_tasks
```

Sample results:

```
$ ansible-playbook play.yml -i ec2.py -vvv
Using /home/lee212/git/aws-cloudformation-by-ansible/ansible.cfg as config file

PLAYBOOK: play.yml *****
1 plays in play.yml

PLAY [create a test instance] *****

TASK [Create security group] *****
task path: /home/lee212/git/aws-cloudformation-by-ansible/play.yml:5
Wednesday 14 September 2016 00:22:33 -0400 (0:00:00.019) 0:00:00.019 ***
<127.0.0.1> ESTABLISH LOCAL CONNECTION FOR USER: lee212
<127.0.0.1> EXEC /bin/sh -c '( umask 77 && mkdir -p "` echo $HOME/.ansible/tmp/
↪ansible-tmp-1473826953.12-47283165924615 `"' && echo ansible-tmp-1473826953.12-
↪47283165924615="` echo $HOME/.ansible/tmp/ansible-tmp-1473826953.12-47283165924615 `
↪" ) && sleep 0'
<127.0.0.1> PUT /tmp/tmpvNc7Ff TO /home/lee212/.ansible/tmp/ansible-tmp-1473826953.12-
↪47283165924615/ec2_group
<127.0.0.1> EXEC /bin/sh -c 'LANG=en_US.UTF-8 LC_ALL=en_US.UTF-8 LC_MESSAGES=en_US.
↪UTF-8 /home/lee212/venv/ansible/bin/python /home/lee212/.ansible/tmp/ansible-tmp-
↪1473826953.12-47283165924615/ec2_group; rm -rf "/home/lee212/.ansible/tmp/ansible-
↪tmp-1473826953.12-47283165924615/" > /dev/null 2>&1 && sleep 0'
ok: [localhost] => {"changed": false, "group_id": "sg-692c4d13", "invocation": {
↪ "module_args": {"aws_access_key": null, "aws_secret_key": null, "description": "A
↪ Security group", "ec2_url": null, "name": "security-group-test", "profile": null,
↪ "purge_rules": true, "purge_rules_egress": true, "region": "us-east-1", "rules": [{
↪ "cidr_ip": "0.0.0.0/0", "from_port": 22, "proto": "tcp", "to_port": 22}], "rules_
↪ egress": [{"cidr_ip": "0.0.0.0/0", "from_port": null, "group_desc": "example of ec2_
↪ secgroup", "proto": "-1", "to_port": null}], "security_token": null, "state": "present
↪", "validate_certs": true, "vpc_id": "vpc-e6c17c83"}, "module_name": "ec2_group"}}
```

```
TASK [ec2 launch test] *****
task path: /home/lee212/git/aws-cloudformation-by-ansible/play.yml:22
Wednesday 14 September 2016 00:22:33 -0400 (0:00:00.552) 0:00:00.571 ***
<127.0.0.1> ESTABLISH LOCAL CONNECTION FOR USER: lee212
<127.0.0.1> EXEC /bin/sh -c '( umask 77 && mkdir -p "` echo $HOME/.ansible/tmp/
↳ansible-tmp-1473826953.7-216570325713589 `"' && echo ansible-tmp-1473826953.7-
↳216570325713589="` echo $HOME/.ansible/tmp/ansible-tmp-1473826953.7-216570325713589_
↳`"' ) && sleep 0'
<127.0.0.1> PUT /tmp/tmpnwiErZ TO /home/lee212/.ansible/tmp/ansible-tmp-1473826953.7-
↳216570325713589/ec2
<127.0.0.1> EXEC /bin/sh -c 'LANG=en_US.UTF-8 LC_ALL=en_US.UTF-8 LC_MESSAGES=en_US.
↳UTF-8 /home/lee212/venv/ansible/bin/python /home/lee212/.ansible/tmp/ansible-tmp-
↳1473826953.7-216570325713589/ec2; rm -rf "/home/lee212/.ansible/tmp/ansible-tmp-
↳1473826953.7-216570325713589/" > /dev/null 2>&1 && sleep 0'
changed: [localhost] => {"changed": true, "instance_ids": ["i-f7364c0f"], "instances
↳": [{"ami_launch_index": "0", "architecture": "x86_64", "block_device_mapping": {""/
↳dev/sda1": {"delete_on_termination": true, "status": "attached", "volume_id": "vol-
↳f678e477"}}, "dns_name": "", "ebs_optimized": false, "groups": {"sg-692c4d13":
↳"security-group-test"}, "hypervisor": "xen", "id": "i-f7364c0f", "image_id": "ami-
↳2d39803a", "instance_type": "t2.micro", "kernel": null, "key_name": "hrlee",
↳"launch_time": "2016-09-14T04:22:34.000Z", "placement": "us-east-1a", "private_dns_
↳name": "ip-172-30-3-157.ec2.internal", "private_ip": "172.30.3.157", "public_dns_
↳name": "", "public_ip": "54.197.111.173", "ramdisk": null, "region": "us-east-1",
↳"root_device_name": "/dev/sda1", "root_device_type": "ebs", "state": "running",
↳"state_code": 16, "tags": {}, "tenancy": "default", "virtualization_type": "hvm"}},
↳"invocation": {"module_args": {"assign_public_ip": true, "aws_access_key": null,
↳"aws_secret_key": null, "count": 1, "count_tag": null, "ebs_optimized": false, "ec2_
↳url": null, "exact_count": null, "group": null, "group_id": ["sg-692c4d13"], "id":
↳null, "image": "ami-2d39803a", "instance_ids": null, "instance_profile_name": null,
↳"instance_tags": null, "instance_type": "t2.micro", "kernel": null, "key_name":
↳"hrlee", "monitoring": false, "network_interfaces": null, "placement_group": null,
↳"private_ip": null, "profile": null, "ramdisk": null, "region": "us-east-1",
↳"security_token": null, "source_dest_check": true, "spot_launch_group": null, "spot_
↳price": null, "spot_type": "one-time", "spot_wait_timeout": "600", "state": "present
↳", "tenancy": "default", "termination_protection": false, "user_data": null,
↳"validate_certs": true, "volumes": null, "vpc_subnet_id": "subnet-719a774d", "wait
↳": true, "wait_timeout": "300", "zone": null}, "module_name": "ec2"}, "tagged_
↳instances": []}
```

```
TASK [Add new instance to host group] *****
task path: /home/lee212/git/aws-cloudformation-by-ansible/play.yml:35
Wednesday 14 September 2016 00:22:55 -0400 (0:00:22.170) 0:00:22.741 ***
creating host via 'add_host': hostname=54.197.111.173
changed: [localhost] => (item={u'kernel': None, u'root_device_type': u'ebs', u
↳'private_dns_name': u'ip-172-30-3-157.ec2.internal', u'public_ip': u'54.197.111.173
↳', u'private_ip': u'172.30.3.157', u'id': u'i-f7364c0f', u'ebs_optimized': False, u
↳'state': u'running', u'virtualization_type': u'hvm', u'architecture': u'x86_64', u
↳'ramdisk': None, u'block_device_mapping': {u'/dev/sda1': {u'status': u'attached', u
↳'delete_on_termination': True, u'volume_id': u'vol-f678e477'}}, u'key_name': u'hrlee
↳', u'image_id': u'ami-2d39803a', u'tenancy': u'default', u'groups': {u'sg-692c4d13
↳': u'security-group-test'}, u'public_dns_name': u'', u'state_code': 16, u'tags': {},
↳ u'placement': u'us-east-1a', u'ami_launch_index': u'0', u'dns_name': u'', u'region
↳': u'us-east-1', u'launch_time': u'2016-09-14T04:22:34.000Z', u'instance_type': u
↳'t2.micro', u'root_device_name': u'/dev/sda1', u'hypervisor': u'xen'}) => {"add_host
↳": {"groups": ["launched"], "host_name": "54.197.111.173", "host_vars": {}},
↳"changed": true, "invocation": {"module_args": {"groupname": "launched", "hostname
↳": "54.197.111.173"}, "module_name": "add_host"}, "item": {"ami_launch_index": "0",
↳"architecture": "x86_64", "block_device_mapping": {""/dev/sda1": {"delete_on_
↳termination": true, "status": "attached", "volume_id": "vol-f678e477"}}, "dns_name
↳": "", "ebs_optimized": false, "groups": {"sg-692c4d13": "security-group-test"},
↳"hypervisor": "xen", "id": "i-f7364c0f", "image_id": "ami-2d39803a", "instance_type
↳": "t2.micro", "kernel": null, "key_name": "hrlee", "launch_time": "2016-09-
↳14T04:22:34.000Z", "placement": "us-east-1a", "private_dns_name": "ip-172-30-3-157.
↳ec2.internal", "private_ip": "172.30.3.157", "public_dns_name": "", "public_ip":
```

```

TASK [Wait for SSH to come up] *****
task path: /home/lee212/git/aws-cloudformation-by-ansible/play.yml:38
Wednesday 14 September 2016 00:22:55 -0400 (0:00:00.035) 0:00:22.776 ***
<127.0.0.1> ESTABLISH LOCAL CONNECTION FOR USER: lee212
<127.0.0.1> EXEC /bin/sh -c '( umask 77 && mkdir -p "` echo $HOME/.ansible/tmp/
↳ansible-tmp-1473826975.89-214245185864373 `"' && echo ansible-tmp-1473826975.89-
↳214245185864373="` echo $HOME/.ansible/tmp/ansible-tmp-1473826975.89-
↳214245185864373 `"' ) && sleep 0'
<127.0.0.1> PUT /tmp/tmp2vGL53 TO /home/lee212/.ansible/tmp/ansible-tmp-1473826975.89-
↳214245185864373/wait_for
<127.0.0.1> EXEC /bin/sh -c 'LANG=en_US.UTF-8 LC_ALL=en_US.UTF-8 LC_MESSAGES=en_US.
↳UTF-8 /home/lee212/venv/ansible/bin/python /home/lee212/.ansible/tmp/ansible-tmp-
↳1473826975.89-214245185864373/wait_for; rm -rf "/home/lee212/.ansible/tmp/ansible-
↳tmp-1473826975.89-214245185864373/" > /dev/null 2>&1 && sleep 0'
ok: [localhost] => (item={u'kernel': None, u'root_device_type': u'efs', u'private_dns_
↳name': u'ip-172-30-3-157.ec2.internal', u'public_ip': u'54.197.111.173', u'private_
↳ip': u'172.30.3.157', u'id': u'i-f7364c0f', u'efs_optimized': False, u'state': u
↳'running', u'virtualization_type': u'hvm', u'architecture': u'x86_64', u'ramdisk':
↳None, u'block_device_mapping': {u'/dev/sda1': {u'status': u'attached', u'delete_on_
↳termination': True, u'volume_id': u'vol-f678e477'}}}, u'key_name': u'hrlee', u'image_
↳id': u'ami-2d39803a', u'tenancy': u'default', u'groups': {u'sg-692c4d13': u
↳'security-group-test'}, u'public_dns_name': u'', u'state_code': 16, u'tags': {}, u
↳'placement': u'us-east-1a', u'ami_launch_index': u'0', u'dns_name': u'', u'region':
↳u'us-east-1', u'launch_time': u'2016-09-14T04:22:34.000Z', u'instance_type': u't2.
↳micro', u'root_device_name': u'/dev/sda1', u'hypervisor': u'xen'}) => {"changed":
↳false, "elapsed": 60, "invocation": {"module_args": {"connect_timeout": 5, "delay":
↳60, "exclude_hosts": null, "host": "54.197.111.173", "path": null, "port": 22,
↳"search_regex": null, "state": "started", "timeout": 320}, "module_name": "wait_for
↳"}, "item": {"ami_launch_index": "0", "architecture": "x86_64", "block_device_
↳mapping": {"/dev/sda1": {"delete_on_termination": true, "status": "attached",
↳"volume_id": "vol-f678e477"}}, "dns_name": "", "efs_optimized": false, "groups": {
↳"sg-692c4d13": "security-group-test"}, "hypervisor": "xen", "id": "i-f7364c0f",
↳"image_id": "ami-2d39803a", "instance_type": "t2.micro", "kernel": null, "key_name
↳": "hrlee", "launch_time": "2016-09-14T04:22:34.000Z", "placement": "us-east-1a",
↳"private_dns_name": "ip-172-30-3-157.ec2.internal", "private_ip": "172.30.3.157",
↳"public_dns_name": "", "public_ip": "54.197.111.173", "ramdisk": null, "region":
↳"us-east-1", "root_device_name": "/dev/sda1", "root_device_type": "efs", "state":
↳"running", "state_code": 16, "tags": {}, "tenancy": "default", "virtualization_type
↳": "hvm"}, "path": null, "port": 22, "search_regex": null, "state": "started"}

PLAY RECAP *****
localhost : ok=4 changed=2 unreachable=0 failed=0

Wednesday 14 September 2016 00:23:56 -0400 (0:01:00.194) 0:01:22.971 ***
=====
Wait for SSH to come up ----- 60.19s
/home/lee212/git/aws-cloudformation-by-ansible/play.yml:38 -----
ec2 launch test ----- 22.17s
/home/lee212/git/aws-cloudformation-by-ansible/play.yml:22 -----
Create security group ----- 0.55s
/home/lee212/git/aws-cloudformation-by-ansible/play.yml:5 -----
Add new instance to host group ----- 0.04s
/home/lee212/git/aws-cloudformation-by-ansible/play.yml:35 -----

```





## CHAPTER 6

---

ec2.py

---

ec2.py --list option displays cached data. Try with --refresh for update.

```
$ ./ec2.py --list
{
  "_meta": {
    "hostvars": {
      "54.197.206.131": {
        "ansible_ssh_host": "54.197.206.131",
        "ec2__in_monitoring_element": false,
        "ec2_ami_launch_index": "0",
        "ec2_architecture": "x86_64",
        "ec2_client_token": "",
        "ec2_dns_name": "",
        "ec2_ebs_optimized": false,
        "ec2_eventsSet": "",
        "ec2_group_name": "",
        "ec2_hypervisor": "xen",
        "ec2_id": "i-b4a8d24c",
        "ec2_image_id": "ami-2d39803a",
        "ec2_instance_profile": "",
        "ec2_instance_type": "t2.micro",
        "ec2_ip_address": "54.197.206.131",
        "ec2_item": "",
        "ec2_kernel": "",
        "ec2_key_name": "hrlee",
        "ec2_launch_time": "2016-09-14T14:52:55.000Z",
        "ec2_monitored": false,
        "ec2_monitoring": "",
        "ec2_monitoring_state": "disabled",
        "ec2_persistent": false,
        "ec2_placement": "us-east-1a",
        "ec2_platform": "",
        "ec2_previous_state": "",
        "ec2_previous_state_code": 0,
        "ec2_private_dns_name": "ip-172-30-3-230.ec2.internal",
```

```

    "ec2_private_ip_address": "172.30.3.230",
    "ec2_public_dns_name": "",
    "ec2_ramdisk": "",
    "ec2_reason": "",
    "ec2_region": "us-east-1",
    "ec2_requester_id": "",
    "ec2_root_device_name": "/dev/sda1",
    "ec2_root_device_type": "ebs",
    "ec2_security_group_ids": "sg-692c4d13",
    "ec2_security_group_names": "security-group-test",
    "ec2_sourceDestCheck": "true",
    "ec2_spot_instance_request_id": "",
    "ec2_state": "running",
    "ec2_state_code": 16,
    "ec2_state_reason": "",
    "ec2_subnet_id": "subnet-719a774d",
    "ec2_virtualization_type": "hvm",
    "ec2_vpc_id": "vpc-e6c17c83"
  }
}
},
"ami_2d39803a": [
  "54.197.206.131"
],
"ec2": [
  "54.197.206.131"
],
"i-b4a8d24c": [
  "54.197.206.131"
],
"key_hrlee": [
  "54.197.206.131"
],
"security_group_security_group_test": [
  "54.197.206.131"
],
"tag_none": [
  "54.197.206.131"
],
"type_t2_micro": [
  "54.197.206.131"
],
"us-east-1": [
  "54.197.206.131"
],
"us-east-1a": [
  "54.197.206.131"
],
"vpc_id_vpc_e6c17c83": [
  "54.197.206.131"
]
}
}

```

You can compare with aws ec2 results:

```

$ aws ec2 describe-instances
{
  "Reservations": [

```

```

{
  "OwnerId": "461335111454",
  "ReservationId": "r-233f22da",
  "Groups": [],
  "Instances": [
    {
      "Monitoring": {
        "State": "disabled"
      },
      "PublicDnsName": "",
      "State": {
        "Code": 16,
        "Name": "running"
      },
      "EbsOptimized": false,
      "LaunchTime": "2016-09-14T14:52:55.000Z",
      "PublicIpAddress": "54.197.206.131",
      "PrivateIpAddress": "172.30.3.230",
      "ProductCodes": [],
      "VpcId": "vpc-e6c17c83",
      "StateTransitionReason": "",
      "InstanceId": "i-b4a8d24c",
      "ImageId": "ami-2d39803a",
      "PrivateDnsName": "ip-172-30-3-230.ec2.internal",
      "KeyName": "hrlee",
      "SecurityGroups": [
        {
          "GroupName": "security-group-test",
          "GroupId": "sg-692c4d13"
        }
      ],
      "ClientToken": "",
      "SubnetId": "subnet-719a774d",
      "InstanceType": "t2.micro",
      "NetworkInterfaces": [
        {
          "Status": "in-use",
          "MacAddress": "06:a5:49:cf:87:a1",
          "SourceDestCheck": true,
          "VpcId": "vpc-e6c17c83",
          "Description": "",
          "Association": {
            "PublicIp": "54.197.206.131",
            "PublicDnsName": "",
            "IpOwnerId": "amazon"
          },
          "NetworkInterfaceId": "eni-8ae97085",
          "PrivateIpAddresses": [
            {
              "Association": {
                "PublicIp": "54.197.206.131",
                "PublicDnsName": "",
                "IpOwnerId": "amazon"
              },
              "Primary": true,
              "PrivateIpAddress": "172.30.3.230"
            }
          ]
        }
      ]
    }
  ],

```

```
        "Attachment": {
            "Status": "attached",
            "DeviceIndex": 0,
            "DeleteOnTermination": true,
            "AttachmentId": "eni-attach-4243efba",
            "AttachTime": "2016-09-14T14:52:55.000Z"
        },
        "Groups": [
            {
                "GroupName": "security-group-test",
                "GroupId": "sg-692c4d13"
            }
        ],
        "SubnetId": "subnet-719a774d",
        "OwnerId": "461335111454",
        "PrivateIpAddress": "172.30.3.230"
    }
],
"SourceDestCheck": true,
"Placement": {
    "Tenancy": "default",
    "GroupName": "",
    "AvailabilityZone": "us-east-1a"
},
"Hypervisor": "xen",
"BlockDeviceMappings": [
    {
        "DeviceName": "/dev/sda1",
        "Ebs": {
            "Status": "attached",
            "DeleteOnTermination": true,
            "VolumeId": "vol-21e37ca0",
            "AttachTime": "2016-09-14T14:52:56.000Z"
        }
    }
],
"Architecture": "x86_64",
"RootDeviceType": "ebs",
"RootDeviceName": "/dev/sda1",
"VirtualizationType": "hvm",
"AmiLaunchIndex": 0
}
]
}
}
```

Dynamic inventory provides metadata from ec2 module and `hostvars` contains additional information like:

```
{u'ansible_all_ipv4_addresses': [u'172.30.3.139'], u'ansible_all_ipv6_addresses': [u'fe80::4b9:ffff:fe81:65f'],
  u'ansible_architecture': u'x86_64', u'ansible_bios_date': u'05/12/2016', u'ansible_bios_version':
  u'4.2.amazon', 'ansible_check_mode': False, u'ansible_cmdline': {u'BOOT_IMAGE': u'/boot/vmlinuz-
  3.13.0-91-generic',
    u'console': u'ttyS0', u'ro': True, u'root': u'UUID=73362e04-90b8-4c53-b878-2097820e0b34'},
  u'ansible_date_time': {u'date': u'2016-09-16', u'day': u'16', u'epoch': u'1473991427', u'hour':
  u'02', u'iso8601': u'2016-09-16T02:03:47Z', u'iso8601_basic': u'20160916T020347397930',
  u'iso8601_basic_short': u'20160916T020347', u'iso8601_micro': u'2016-09-16T02:03:47.398008Z',
  u'minute': u'03', u'month': u'09', u'second': u'47', u'time': u'02:03:47', u'tz': u'UTC', u'tz_offset':
  u'+0000', u'weekday': u'Friday', u'weekday_number': u'5', u'weeknumber': u'37', u'year': u'2016'},
  u'ansible_default_ipv4': {u'address': u'172.30.3.139', u'alias': u'eth0', u'broadcast': u'172.30.3.255',
  u'gateway': u'172.30.3.1', u'interface': u'eth0', u'macaddress': u'06:b9:ff:81:06:5f', u'mtu': 9001,
  u'netmask': u'255.255.255.0', u'network': u'172.30.3.0', u'type': u'ether'},
  u'ansible_default_ipv6': {}, u'ansible_devices': {u'xvda': {u'holders': [],
    u'host': u'', u'model': None, u'partitions': {u'xvda1': {u'sectors': u'16755795',
      u'sectorsize': 512, u'size': u'7.99 GB', u'start': u'16065'}}},
    u'removable': u'0', u'rotational': u'0', u'sas_address': None, u'sas_device_handle': None,
    u'scheduler_mode': u'deadline', u'sectors': u'16777216', u'sectorsize': u'512', u'size': u'8.00 GB',
    u'support_discard': u'0', u'vendor': None}},
  u'ansible_distribution': u'Ubuntu', u'ansible_distribution_major_version': u'14',
  u'ansible_distribution_release': u'trusty', u'ansible_distribution_version': u'14.04', u'ansible_dns':
  {u'nameservers': [u'172.30.0.2'],
    u'search': [u'ec2.internal']},
  u'ansible_domain': u'', u'ansible_env': {u'HOME': u'/home/ubuntu',
```

```

u'LANG': u'en_US.UTF-8', u'LC_ALL': u'en_US.UTF-8', u'LC_MESSAGES':
u'en_US.UTF-8', u'LOGNAME': u'ubuntu', u'MAIL': u'/var/mail/ubuntu', u'PATH':
u'/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games',
u'PWD': u'/home/ubuntu', u'SHELL': u'/bin/bash', u'SHLVL': u'1', u'SSH_CLIENT':
u'156.56.102.202 38016 22', u'SSH_CONNECTION': u'156.56.102.202 38016 172.30.3.139 22',
u'SSH_TTY': u'/dev/pts/0', u'TERM': u'xterm', u'USER': u'ubuntu', u'XDG_RUNTIME_DIR':
u'/run/user/1000', u'XDG_SESSION_ID': u'8', u'_': u'/bin/sh'},

u'ansible_eth0': {u'active': True, u'device': u'eth0, u'ipv4': {u'address': u'172.30.3.139',
    u'broadcast': u'172.30.3.255', u'netmask': u'255.255.255.0', u'network': u'172.30.3.0'},
u'ipv6': [{u'address': u'fe80::4b9:ffff:fe81:65f', u'prefix': u'64', u'scope': u'link'}],
    u'macaddress': u'06:b9:ff:81:06:5f', u'mtu': 9001, u'pciid': u'vif-0', u'promisc': False, u'type':
    u'ether'},

u'ansible_fips': False, u'ansible_form_factor': u'Other', u'ansible_fqdn': u'ip-172-30-3-139',
u'ansible_gather_subset': [u'hardware', u'network', u'virtual'], u'ansible_hostname': u'ip-172-30-3-139',
u'ansible_interfaces': [u'lo', u'eth0'], u'ansible_kernel': u'3.13.0-91-generic', u'ansible_lo': {u'active': True,
    u'device': u'lo', u'ipv4': {u'address': u'127.0.0.1',
        u'broadcast': u'host', u'netmask': u'255.0.0.0', u'network': u'127.0.0.0'},
u'ipv6': [{u'address': u'::1', u'prefix': u'128', u'scope': u'host'}],
    u'mtu': 65536, u'promisc': False, u'type': u'loopback'},

u'ansible_lsb': {u'codename': u'trusty', u'description': u'Ubuntu 14.04.4 LTS', u'id': u'Ubuntu',
    u'major_release': u'14', u'release': u'14.04'},

u'ansible_machine': u'x86_64', u'ansible_machine_id': u'dd1bac9cb23792a632bd6ee157719517',
u'ansible_memfree_mb': 715, u'ansible_memory_mb': {u'nocache': {u'free': 932, u'used': 60},
    u'real': {u'free': 715, u'total': 992, u'used': 277}, u'swap': {u'cached': 0,
        u'free': 0, u'total': 0, u'used': 0}},

u'ansible_memtotal_mb': 992, u'ansible_mounts': [{u'device': u'/dev/xvda1',
    u'fstype': u'ext4', u'mount': u '/', u'options': u'rw,discard', u'size_available': 7029567488,
    u'size_total': 8309932032, u'uuid': u''}],

u'ansible_nodename': u'ip-172-30-3-139', u'ansible_os_family': u'Debian', u'ansible_pkg_mgr': u'apt',
u'ansible_processor': [u'GenuineIntel',
    u'Intel(R) Xeon(R) CPU E5-2670 v2 @ 2.50GHz'],

u'ansible_processor_cores': 1, u'ansible_processor_count': 1, u'ansible_processor_threads_per_core': 1,
u'ansible_processor_vcpus': 1, u'ansible_product_name': u'HVM domU', u'ansible_product_serial':
u'NA', u'ansible_product_uuid': u'NA', u'ansible_product_version': u'4.2.amazon', u'ansible_python':
{u'executable': u'/usr/bin/python',
    u'has_sslcontext': False, u'type': u'CPython', u'version': {u'major': 2,
        u'micro': 6, u'minor': 7, u'releaselevel': u'final', u'serial': 0},
    u'version_info': [2, 7, 6, u'final', 0]},

u'ansible_python_version': u'2.7.6', u'ansible_selinux': False, u'ansible_service_mgr':
u'upstart', u'ansible_ssh_host': u'52.23.213.103', u'ansible_ssh_host_key_dsa_public':
u'AAAAB3NzaC1kc3MAAACBAKosQQsY22MWjQv/bF7V6kewbM5Z7NIPjYgNL/j8IbUrmZv/kZSAArfs/ByKjC2gesSZUi3I
u'ansible_ssh_host_key_ecdsa_public': u'AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBBCjuapv
u'ansible_ssh_host_key_ed25519_public': u'AAAAC3NzaC1lZDI1NTE5AAAAIEOzako99Tn5mO/vjk+ctHc9HKmtdsdJxTqqN

```

```

u'ansible_ssh_host_key_rsa_public': u'AAAAB3NzaC1yc2EAAAADAQABAAQAC7elCMaVXdqIuBrO+4Fx5oPggQdI7JF
u'ansible_swapfree_mb': 0, u'ansible_swaptotal_mb': 0, u'ansible_system': u'Linux',
u'ansible_system_capabilities': [u''], u'ansible_system_capabilities_enforced': u'True',
u'ansible_system_vendor': u'Xen', u'ansible_uptime_seconds': 21477, u'ansible_user_dir': u'/home/ubuntu',
u'ansible_user_gecos': u'Ubuntu', u'ansible_user_gid': 1000, u'ansible_user_id': u'ubuntu',
u'ansible_user_shell': u'/bin/bash', u'ansible_user_uid': 1000, u'ansible_userspace_architecture': u'x86_64',
u'ansible_userspace_bits': u'64', 'ansible_version': {'full': '2.1.1.0',

```

```

'major': 2, 'minor': 1, 'revision': 1, 'string': '2.1.1.0'},

```

```

u'ansible_virtualization_role': u'guest', u'ansible_virtualization_type': u'xen',
u'ec2__in_monitoring_element': False, u'ec2_ami_launch_index': u'0', u'ec2_architecture':
u'x86_64', u'ec2_client_token': u'', u'ec2_dns_name': u'', u'ec2_ebs_optimized': False,
u'ec2_eventsSet': u'', u'ec2_group_name': u'', u'ec2_hypervisor': u'xen', u'ec2_id': u'i-435014bb',
u'ec2_image_id': u'ami-2d39803a', u'ec2_instance_profile': u'', u'ec2_instance_type': u't2.micro',
u'ec2_ip_address': u'52.23.213.103', u'ec2_item': u'', u'ec2_kernel': u'', u'ec2_key_name': u'albert',
u'ec2_launch_time': u'2016-09-15T20:05:34.000Z', u'ec2_monitored': False, u'ec2_monitoring':
u'', u'ec2_monitoring_state': u'disabled', u'ec2_persistent': False, u'ec2_placement': u'us-
east-1a', u'ec2_platform': u'', u'ec2_previous_state': u'', u'ec2_previous_state_code': 0,
u'ec2_private_dns_name': u'ip-172-30-3-139.ec2.internal', u'ec2_private_ip_address': u'172.30.3.139',
u'ec2_public_dns_name': u'', u'ec2_ramdisk': u'', u'ec2_reason': u'', u'ec2_region': u'us-east-
1', u'ec2_requester_id': u'', u'ec2_root_device_name': u'/dev/sda1', u'ec2_root_device_type':
u'ebs', u'ec2_security_group_ids': u'sg-69ebc60c,sg-115b376b', u'ec2_security_group_names':
u'default,abds_secgroup', u'ec2_sourceDestCheck': u'true', u'ec2_spot_instance_request_id': u'',
u'ec2_state': u'running', u'ec2_state_code': 16, u'ec2_state_reason': u'', u'ec2_subnet_id': u'subnet-
719a774d', u'ec2_tag_datanodes': u'True', u'ec2_tag_frontendnodes': u'True', u'ec2_tag_hadoopnodes':
u'True', u'ec2_tag_historyservernodes': u'True', u'ec2_tag_journalnodes': u'True', u'ec2_tag_namenodes':
u'True', u'ec2_tag_resourcemanagernodes': u'True', u'ec2_tag_zookeepernodes': u'True',
u'ec2_virtualization_type': u'hvm', u'ec2_vpc_id': u'vpc-e6c17c83', 'group_names': [u'ami_2d39803a',

```

```

u'ec2', u'i-435014bb', u'key_albert', u'security_group_abds_secgroup', u'security_group_default',
u'tag_datanodes_True', u'tag_frontendnodes_True', u'tag_hadoopnodes_True',
u'tag_historyservernodes_True', u'tag_journalnodes_True', u'tag_namenodes_True',
u'tag_resourcemanagernodes_True', u'tag_zookeepernodes_True', u'type_t2_micro', u'us-east-1',
u'us-east-1a', u'vpc_id_vpc_e6c17c83'],

```

```

'groups': {'all': [u'52.23.213.103', u'54.196.41.145', u'54.209.137.235'],

```

```

u'ami_2d39803a': [u'52.23.213.103', u'54.196.41.145', u'54.209.137.235'],

```

```

u'ec2': [u'52.23.213.103', u'54.196.41.145', u'54.209.137.235'], u'i-435014bb': [u'52.23.213.103'], u'i-
e250141a': [u'54.196.41.145'], u'i-e350141b': [u'54.209.137.235'], u'key_albert': [u'52.23.213.103',

```

```

u'54.196.41.145', u'54.209.137.235'],

```

```

u'security_group_abds_secgroup': [u'52.23.213.103', u'54.196.41.145', u'54.209.137.235'],

```

```

u'security_group_default': [u'52.23.213.103', u'54.196.41.145', u'54.209.137.235'],

```

```

u'tag_datanodes_True': [u'52.23.213.103', u'54.196.41.145', u'54.209.137.235'],

```

```

u'tag_frontendnodes_False': [u'54.196.41.145', u'54.209.137.235'], u'tag_frontendnodes_True':
[u'52.23.213.103'], u'tag_hadoopnodes_True': [u'52.23.213.103',

```

```

u'54.196.41.145', u'54.209.137.235'],

```

```

u'tag_historyservernodes_False': [u'54.196.41.145', u'54.209.137.235'],

```

```

u'tag_historyservernodes_True': [u'52.23.213.103'], u'tag_journalnodes_True': [u'52.23.213.103',

```

```

u'54.196.41.145', u'54.209.137.235'],

```

```
u'tag_namenodes_True': [u'52.23.213.103', u'54.196.41.145', u'54.209.137.235'],
u'tag_resourcemanagernodes_True': [u'52.23.213.103', u'54.196.41.145', u'54.209.137.235'],
u'tag_zookeepernodes_True': [u'52.23.213.103', u'54.196.41.145', u'54.209.137.235'],
u'type_t2_micro': [u'52.23.213.103', u'54.196.41.145', u'54.209.137.235'],
'ungrouped': ['localhost'], u'us-east-1': [u'52.23.213.103',
    u'54.196.41.145', u'54.209.137.235'],
u'us-east-1a': [u'52.23.213.103', u'54.196.41.145', u'54.209.137.235'],
u'vpc_id_vpc_e6c17c83': [u'52.23.213.103', u'54.196.41.145', u'54.209.137.235']],
'inventory_dir': '/home/albert/git/aws-cloudformation-by-ansible', 'inventory_file': 'ec2.py', 'in-
ventory_hostname': u'52.23.213.103', 'inventory_hostname_short': u'52', u'module_setup': True,
'omit': '__omit_place_holder__943b1cb4d91b305bcddb59adcbdfd6146fda5e7e', 'playbook_dir':
u'/home/albert/git/aws-cloudformation-by-ansible'}
```

Documentation about ec2 facts: [http://docs.ansible.com/ansible/ec2\\_facts\\_module.html](http://docs.ansible.com/ansible/ec2_facts_module.html)

## In Practice

`boot.yml` launches ec2 instances and `ec2.py --list` returns running ec2 instances with public ip addresses. When the actual playbook `site.yml` runs for Big Data Stacks, inventory groups and IPs are required to connect.

## Traditional Inventory file

`inventory.txt` contains, for example:

```
[namenodes]
mycluster0
mycluster1

[resourcemanagernodes]
mycluster0
mycluster1

[datanodes]
mycluster0
mycluster1
mycluster2

[zookeepernodes]
mycluster0
mycluster1
mycluster2

[hadoopnodes]
mycluster0
mycluster1
mycluster2

[historyservernodes]
mycluster2
```



```
[journalnodes]
mycluster2
mycluster1
mycluster0

[frontendnodes]
mycluster2
```

and the following directories for each hostname, e.g. mycluster[0-2] look like:

```
$ cat host_vars/mycluster0
ansible_ssh_host: 192.168.1.100
zookeeper_id: 0
```

## Dynamic Inventory (ec2)

Let's assume the official `ec2.py` is used for dynamic inventory for Amazon EC2. (It's from <https://raw.githubusercontent.com/ansible/ansible/devel/contrib/inventory/ec2.py>)

It returns JSON based on running instances on Amazon, for example, if no vm is running:

```
$ ./ec2.py --list
{
  "_meta": {
    "hostvars": {}
  }
}
```

**Note:** `chmod a+x ec2.py` for executable

If there is running VMs, the return JSON data looks like:

```
{
  "_meta": {
    "hostvars": {
      "54.85.130.171": { "ansible_ssh_host": "54.85.130.171", "ec2_in_monitoring_element": false,
        "ec2_ami_launch_index": "0", "ec2_architecture": "x86_64", "ec2_client_token": "",
        "ec2_dns_name": "", "ec2_ebs_optimized": false, "ec2_eventsSet": "", "ec2_group_name":
        "", "ec2_hypervisor": "xen", "ec2_id": "i-1a2c68e2", "ec2_image_id": "ami-2d39803a",
        "ec2_instance_profile": "", "ec2_instance_type": "t2.micro", "ec2_ip_address":
        "54.85.130.171", "ec2_item": "", "ec2_kernel": "", "ec2_key_name": "albert",
        "ec2_launch_time": "2016-09-15T17:18:36.000Z", "ec2_monitored": false, "ec2_monitoring":
        "", "ec2_monitoring_state": "disabled", "ec2_persistent": false, "ec2_placement": "us-
        east-1a", "ec2_platform": "", "ec2_previous_state": "", "ec2_previous_state_code":
        0, "ec2_private_dns_name": "ip-172-30-3-104.ec2.internal", "ec2_private_ip_address":
        "172.30.3.104", "ec2_public_dns_name": "", "ec2_ramdisk": "", "ec2_reason":
        "", "ec2_region": "us-east-1", "ec2_requester_id": "", "ec2_root_device_name":
        "/dev/sda1", "ec2_root_device_type": "ebs", "ec2_security_group_ids": "sg-
        69ebc60c,sg-115b376b", "ec2_security_group_names": "default,abds_secgroup",
        "ec2_sourceDestCheck": "true", "ec2_spot_instance_request_id": "", "ec2_state": "run-
        ning", "ec2_state_code": 16, "ec2_state_reason": "", "ec2_subnet_id": "subnet-719a774d",
        "ec2_tag_datanodes": "True", "ec2_tag_frontendnodes": "False", "ec2_tag_hadoopnodes":
```

```
        "True", "ec2_tag_historyservernodes": "False", "ec2_tag_journalnodes": "True",
        "ec2_tag_namenodes": "True", "ec2_tag_resourcemanagernodes": "True",
        "ec2_tag_zookeepernodes": "True", "ec2_virtualization_type": "hvm", "ec2_vpc_id": "vpc-
        e6c17c83"
    }
}
}, "ami_2d39803a": [
    "54.85.130.171"
], "ec2": [
    "54.85.130.171"
], "i-1a2c68e2": [
    "54.85.130.171"
], "key_albert": [
    "54.85.130.171"
], "security_group_abds_secgroup": [
    "54.85.130.171"
], "security_group_default": [
    "54.85.130.171"
], "tag_datanodes_True": [
    "54.85.130.171"
], "tag_frontendnodes_False": [
    "54.85.130.171"
], "tag_hadoopnodes_True": [
    "54.85.130.171"
], "tag_historyservernodes_False": [
    "54.85.130.171"
], "tag_journalnodes_True": [
    "54.85.130.171"
], "tag_namenodes_True": [
    "54.85.130.171"
], "tag_resourcemanagernodes_True": [
    "54.85.130.171"
], "tag_zookeepernodes_True": [
    "54.85.130.171"
], "type_t2_micro": [
    "54.85.130.171"
], "us-east-1": [
```

```

    "54.85.130.171"
  ], "us-east-1a": [
    "54.85.130.171"
  ], "vpc_id_vpc_e6c17c83": [
    "54.85.130.171"
  ]
}

```

## EC2 Tags for Inventory Groups

If you noticed that there are tags for host groups like `tag_hadoopnodes_True`, ec2 instances have key:value tags and these are used for identifying inventory groups.

Ansible `ec2` module provides `instance_tags` option to define them.

Example:

```

ec2:
  ...
  instance_tags:
    namenodes: True
    ...
    frontendnodes: False

```

`boot.yml` defines tags information when it starts EC2 instances.

## add\_host

This task converts `tag_*_True` list to Inventory groups therefore existing roles run without changes between dynamic and static file inventories.

`convert.yml` contains `add_host` tasks to convert the inventory groups.

For example, to convert `hadoopnodes`:

```

- name: convert EC2 tags_*_True inventory groups
  hosts: "tag_hadoopnodes_True"
  tasks:
    - name: Add new instance to host group (hadoopnodes)
      add_host: hostname="{{ hostvars[item]['ec2_ip_address'] }}"
      ↪groupname=hadoopnodes
      with_items: "{{ groups['tag_hadoopnodes_True'] }}"

```

The hosts in the `tag_hadoopnodes_True` list run the `add_host` task to add its `ec2_ip_address` (in this case, AWS VPC, virtual private cloud, associates with a floating ip address and it is visible via `ec2_ip_address`) to the new group name `hadoopnodes` which the following ansible roles for BigData Stack requires.

## site.yml with Include

Three Ansible playbooks are provided to deploy virtual clusters on Amazon EC2.

- `boot.yml`: to start EC2 instances
- `convert.yml`: to provide inventory groups from dynamic inventory of ec2.
- `example-project-nist-fingerprint-matching.yml`: to run SDS playbooks

---

## Facts for Create and Terminate Instances

---

Ansible provides additional EC2 information of `ec2` module using `ec2_facts` which contains instance ids, image ids, regions, etc. Boot up plays creates `ec2` instances for projects and applications are installed and datasets are loaded to complete jobs on Cloud resources. If all jobs are finished successfully, the last step would be terminating VM instances. `boot.yml` playbook stores `ec2` instance ids using `ec2_facts` in YAML format and `terminate.yml` playbook uses the information to delete instances.

Sample `ec2` facts are: `.. include:: ec2_facts_for_vars.txt`

### **to\_nice\_yaml**

`boot.yml` playbook stores instance information in a YAML file therefore `terminate.yml` can find instance ids to terminate. `to_nice_yaml` and `copy` module (with `content` option) is used to store `ec2_fact` information in a YAML file.

### **include\_vars**

The nice thing about Ansible is that data variables in a YAML file can be easily imported using `include_vars`. `terminate.yml` playbook uses it to read the instance information that started by `boot.yml`



## CHAPTER 9

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`