**1. How to reboot Linux nodes in reliable way using ansible?**

- hosts: jdedev-linux

sudo: yes

tasks:

- name: Rebooting server

command: /sbin/reboot

- name: Wait for the server to finish rebooting

local\_action: wait\_for host="{{ inventory\_hostname }}" search\_regex=OpenSSH port=22 timeout=300

WAIT\_FOR MODULE UTILITY :

- name: Wait 300 seconds for port 22 to become open and contain "OpenSSH"

wait\_for:

port: 22

host: '{{ (ansible\_ssh\_host|default(ansible\_host))|default(inventory\_hostname) }}'

search\_regex: OpenSSH

delay: 10

connection: local

WAIT\_FOR\_CONNECTION

# wait\_for\_connection - Waits until remote system is reachable/usable

*# Wake desktops, wait for them to become ready and continue playbook*

- hosts: all

gather\_facts: no

tasks:

- name: Send magic Wake-On-Lan packet to turn on individual systems

wakeonlan:

mac: '{{ mac }}'

broadcast: 192.168.0.255

delegate\_to: localhost

- name: Wait for system to become reachable

wait\_for\_connection:

- name: Gather facts for first time

setup:

**2. BLOCKINFILE MODULE**

Ansible blockinfile module is used to insert/update/remove a block of lines. The block will be surrounded by a marker, like begin and end, to make the task idempotent.

- hosts: blocks

tasks:

- name: Using Ansible blockinfile with\_items

blockinfile:

dest: /home/mdtutorials2/block\_output.txt

block: |

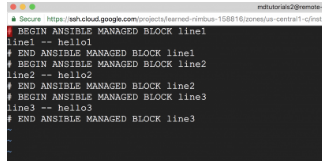
{{ item.name }} -- {{ item.line }}

marker: "# {mark} ANSIBLE MANAGED BLOCK {{ item.name }}"

with\_items:

- { name: line1, line: hello1 }

- { name: line2, line: hello2 }

- { name: line3, line: hello3 }

marker: <!-- {mark} Adding IP address -->

- hosts: blocks

tasks:

- name: Removing a line using blockinfile

blockinfile:

dest: /home/mdtutorials2/block\_output.txt

marker: <!-- {mark} Adding IP address -->

state: absent

Inserting contents from an external file

You can also use the contents from an external file using the [lookup plugin](http://docs.ansible.com/ansible/playbooks_lookups.html). The file should be available in your Ansible control machine.

block: "{{ lookup('file', '/home/user/localfile.txt') }}"

**lineinfile**

- lineinfile:

path: /etc/sudoers

state: present

regexp: '^%ADMIN ALL='

line: '%ADMIN ALL=(ALL) NOPASSWD: ALL'

validate: '/usr/sbin/visudo -cf %s'

- lineinfile:

path: /etc/httpd/conf/httpd.conf

regexp: '^Listen '

insertafter: '^#Listen '

line: 'Listen 8080'

**3. PROMPT MODULE WITH VARIABLE SECTION:**

- hosts: all

gather\_facts: False

vars\_prompt:

- name: "Version"

prompt: "Which version Do you want to install?"

tasks:

- name: Ansible prompt example.

debug:

msg: "{{ Version }}"

Note Sometimes you may need the user to see what he is giving as the input. In that case, you set the private option to ‘no’.

**4. ARCHIVE MODULE:**

Zip Multiple Files

- hosts: all

tasks:

- name: Ansible zip multiple files example

archive:

path:

- /Users/mdtutorials2/Documents/Ansible/zipfile.txt

- /Users/mdtutorials2/Documents/Ansible/zipfile2.txt

- /Users/mdtutorials2/Documents/Ansible/zipfile3.txt

dest: /Users/mdtutorials2/Documents/multi.zip

format: zip

Zip directory

- hosts: all

tasks:

- name: Ansible zip directory example

archive:

path:

- /Users/mdtutorials2/Documents/Ansible/test1

dest: /Users/mdtutorials2/Documents/Ansible/direc.zip

format: zip

**unarchive**

- name: Extract foo.tgz into /var/lib/foo

unarchive:

src: foo.tgz

dest: /var/lib/foo

- name: Unarchive a file that is already on the remote machine

unarchive:

src: /tmp/foo.zip

dest: /usr/local/bin

remote\_src: yes

- name: Unarchive a file that needs to be downloaded (added in 2.0)

unarchive:

src: https://example.com/example.zip

dest: /usr/local/bin

remote\_src: yes

**5. TIME STAMP FACTS**

tasks:

- name: Ansible date fact example

debug:

var=ansible\_date\_time.date

Local Time stamp

- hosts: all

tasks:

- debug: msg="{{ lookup('pipe','date') }}"

6**. SPLIT AND JOIN**

- hosts: all

vars:

test: "This single line should be split based on white space"

tasks:

- debug:

msg: "{{ test.split() }}"

- hosts: all

vars:

test: ['192','168','14','21']

tasks:

- debug:

msg: "{{ test | join('.') }}"

output

------

ok: [localhost] => {

"msg": "192.168.14.21"

}

7. USER & GROUP MODULE WITH PASSWORD HASH

- hosts: all

tasks:

- name: Ansible create user example.

user:

name: test2

password: $6$53uLlaE42OzJi8k1$nah2F17o2XYc1rMg1bZwlIY2XRdgnYrqwVGUJC

bsnpwtqCAGxyn9eN/RRdZNugHbcOLjz/.y4Slou4ut7yl3P0

mkpasswd --method=sha-512

pip install passlib

python -c "from passlib.hash import sha512\_crypt; import getpass; print(sha512\_crypt.using(rounds=5000).hash(getpass.getpass()))"

- hosts: all

tasks:

- name: Ansible append user to group example.

user:

name: test3

groups: admin, googlecloud

append: yes

Group Module example:

- name: Ensure group "somegroup" exists

group:

name: somegroup

state: present

8. REGISTER OUTPUT IN ANSIBLE

- hosts: all

tasks:

- name: Ansible register variable basic example

shell: "find \*.txt"

args:

chdir: "/Users/mdtutorials2/Documents/Ansible"

register: find\_output

- debug:

var: find\_output

output

======

ok: [localhost] => {

"find\_output": {

"changed": true,

"cmd": "find \*.txt",

"delta": "0:00:00.008597",

"end": "2017-09-30 15:07:15.940235",

"rc": 0,

"start": "2017-09-30 15:07:15.931638",

"stderr": "",

"stderr\_lines": [],

"stdout": "check.txt\ncheck2.txt",

"stdout\_lines": [

"check.txt",

"check2.txt"

]

}

- hosts: all

tasks:

- name: Ansible register with\_items example

shell: "find \*.txt"

args:

chdir: "/Users/mdtutorials2/Documents/Ansible"

register: with\_output

- shell: "cp {{ item }} {{item}}\_bkp"

with\_items:

- "{{ with\_output.stdout\_lines }}"

9. FETCH MODULE:

# Fetches a file from remote nodes

*# Specifying a destination path*

- fetch:

src: /tmp/uniquefile

dest: /tmp/special/

flat: yes

10. FILE MODULE

Sets attributes of files, symlinks, and directories, or removes

Create file

- file:

path: /etc/foo.conf

owner: foo

group: foo

mode: 0644

- file:

path: /etc/foo.conf

state: touch

mode: "u+rw,g-wx,o-rwx"

Link file

- file:

src: /file/to/link/to

dest: /path/to/symlink

owner: foo

group: foo

state: link

create directory

- file:

path: /etc/some\_directory

state: directory

mode: 0755

11. DEBUG

# debug - Print statements during execution

- debug:

msg: "System {{ inventory\_hostname }} has uuid {{ ansible\_product\_uuid }}"

- name: Display all variables/facts known for a host

debug:

var: hostvars[inventory\_hostname]

verbosity: 4

12. AUTHORIZED\_KEYS FILE ADDTION

- name: Set authorized key taken from file

authorized\_key:

user: charlie

state: present

key: "{{ lookup('file', '/home/charlie/.ssh/id\_rsa.pub') }}"

12. MAKES FILE SYSTEM:

- name: Create a ext2 filesystem on /dev/sdb1

filesystem:

fstype: ext2

dev: /dev/sdb1

- name: Create a ext4 filesystem on /dev/sdb1 and check disk blocks

filesystem:

fstype: ext4

dev: /dev/sdb1

opts: -cc

12. MOUNT FILE SYSTEM

13. MANAGE HOSTNAME

- hostname:

name: web0