**DevOps**

AnsiblePlaybook

Contents

ACL Module 1

Apache 2 Module 1

Apt Module 2 apt \_repository Module/Apt\_Key Example 2

AT Module 3 kernal\_blacklist Module 3

Command Module 3

Copy Module 4

Cron Module 4

Debug Module 4

Delegate to Function Demo 5 dnf Module 5

Error Handling 6

Fetch Module 6

Filesystem Module 7

Variables at Command Line Passing 7 get\_url Module 7

Git Module 8

Group Module 8

Hostname Module 8 htpasswd Module 9

Full Include Tasks 9

Local Action Playbook 9

Local Action Demo 10

Lookup Playbook 10

Loop Playbook Examples 10

Mail Module 11 modprobe Module 11

Mount Module 12

My First YAML Playbook 12 mysql\_db Module 13 mysql\_user Module 13

Package Module 13

Pause Module 14

Ping Module 14

Prompt for User Package Example 15

Raw Module 15 run\_once Playbook Example 16

Script Module 16

SELinux Module 17

Service Module 17 set\_fact Module 18

Shell Module 18

Start At Playbook Example 19

Stat Module 19

Tag Functionality 20

Unarchive Module 20

Until Example 21

User Module 21 wait\_for Module 22

When Playbook Example 22

Yum Module 23

**ACL Module**

--- # ACL MODULE EXAMPLE

- hosts: apacheweb user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Get ACL Information on the /etc/test.acl.txt remote file

acl: name=/etc/test.acl.txt entity=test etype=user permissions=”rw” state=present register: aclinfo

- debug: var=aclinfo

**Apache 2 Module**

--- # APACHE2\_MODULE EXAMPLE

- hosts: aptserver user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Disable the alias module in Apache2 apache2\_module: state=present name=alias

- hosts: appserver vars:

author\_name: Test user

vars\_files:

- vars.yml tasks:

- name: Install Lynx on App Servers

yum: pkg=lynx state=installed update\_cache=true

**Apt Module**

--- # APT MODULE EXAMPLE

- hosts: aptserver user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Install Apache Web Server

apt: name=apache2 state=present update\_cache=yes

**apt \_repository Module/Apt\_Key Example**

--- # APT\_REPOSITORY MODULE EXAMPLE/ALSO APT\_KEY EXAMPLE

- hosts: aptserver user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Install a dependency needed for apt\_repository apt: pkg=python-apt state=latest

- name: Add the key

apt\_key: url=https://dl-ssl.google.com/linux/linux\_signing\_key.pub state=present

- name: Add the Google Repo for Ubuntu

apt\_repository: repo=’deb [http://dl.google.com/linux/deb/ stable main non-free’ state=present](http://dl.google.com/linux/deb/)

---

- hosts: apacheweb user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Install Apache Web Server

action: yum name=httpd state=installed

- fail: msg=”Installation Failed, this is not CentOS or RedHat Host”

when: “ansible\_os\_family != ‘RedHat’”

**AT Module**

--- # AT MODULE EXAMPLE

- hosts: apacheweb user: test

sudo: sudo connection: ssh gather\_facts: no tasks:

- name: Example of a future command with the AT module

at: command=”ls /var/log > /home/test/at1.log” state=absent

**kernal\_blacklist Module**

--- # KERNEL\_BLACKLIST MODULE DEMO

- hosts: apacheweb user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Blacklist the DUMMY kernel module kernel\_blacklist: name=dummy state=absent

**Command Module**

--- # COMMAND MODULE EXAMPLE

- hosts: appserver user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Check for python packages command: /home/test/testing/test.sh args:

chdir: /home/test/testing

**Copy Module**

--- # COPY MODULE EXAMPLE

- hosts: apacheweb user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Copy from the files directory test file

action: copy src=files/test4.txt dest=/home/test/test4.txt owner=test group=test mode=0655

backup=yes

**Cron Module**

--- # CRON MODULE EXAMPLE

- hosts: apacheweb user: test connection: ssh gather\_facts: no tasks:

- name: Add a CRON Job to the Test User

cron: name=”list dirs” minute=”0” hour=”1” job=”ls -al /var/log > /home/test/cron.log”

**Debug Module**

--- # DEBUG MODULE EXAMPLE

- hosts: apacheweb user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Install web server

yum: name=httpd state=installed

- debug: msg=”Equivalent of sudo yum install httpd”

- name: How Long has the system been up?

shell: /usr/bin/uptime register: result

- debug: var=result

**Delegate to Function Demo**

--- # DELEGATE TO FUNCTION DEMO

- hosts: apacheweb sudo: yes

user: test connection: ssh tasks:

- name: Run a remote ping on the application server raw: ping -c 4 tcox5 > /home/test/Playbooks/ping.out delegate\_to: 127.0.0.1

- name: Install a package

yum: pkg=lynx state=latest

- hosts: appserver tasks:

- name: Install Lynx on App Servers

yum: pkg=lynx state=installed update\_cache=true

- name: Querying for Telnet Install

yum: pkg=telnet state=present update\_cache=true

- hosts: apacheweb tasks:

- name: Install Lynx on Web Servers

yum: pkg=telnet state=installed update\_cache=true

- name: Querying for Lynx Install

yum: pkg=lynx state=present update\_cache=true

**dnf Module**

--- # DNF MODULE EXAMPLE

- hosts: apacheweb user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: DNF Update

dnf: [name=”@Development tools” state=present](mailto:@Development)

---

- hosts: appserver

user: test sudo: yes connection: ssh gather\_facts: no tasks:

- name: Load dummy module

modprobe: name=dummy state=absent

**Error Handling**

--- # ERROR HANDLING EXAMPLE

- hosts: apacheweb user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Execute a command that will fail command: /bin/false

ignore\_errors: yes

- name: Install telnet

yum: pkg=telnet state=latest

**Fetch Module**

--- # FETCH MODULE EXAMPLE

- hosts: apacheweb user: test

sudo: yes connection: ssh tasks:

- name: Copy remote hosts file to control server

fetch: src=/etc/hosts dest=/home/test/prefix-{{ ansible\_hostname }} flat=yes

**Filesystem Module**

--- # FILESYSTEM MODULE EXAMPLE

- hosts: appserver user: test

sudo: yes connection: ssh gather\_facts: no

tasks:

- name: Format the remote data partition

filesystem: fstype=ext3 dev=/dev/xvdf1

**Variables at Command Line Passing**

--- # VARIABLES AT A COMMAND LINE PASSING EXAMPLE

- hosts: ‘{{ hosts }}’ user: ‘{{ user }}’ sudo: yes connection: ssh gather\_facts: no tasks:

- name: Install some software

yum: pkg={{ pkg }} state=latest

**get\_url Module**

--- # GET\_URL MODULE EXAMPLE

- hosts: aptserver user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Get and download the INI file from the web server

get\_url: [url=http://tcox1.mylabserver.com/mytest.ini dest=/home/test/mytest.ini mode=0440](http://tcox1.mylabserver.com/mytest.ini)

**Git Module**

--- # GIT MODULE EXAMPLE

- hosts: apacheweb user: test connection: ssh gather\_facts: no tasks:

- name: Checking out a git repo on the remote server raw: date

**Group Module**

--- # GROUP MODULE EXAMPLE

- hosts: apacheweb user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Add a new group called newgroup group: name=newgroup state=absent

**Hostname Module**

--- # HOSTNAME MODULE EXAMPLE

- hosts: aptserver user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Change the hostname to something else

hostname: name=tcox01

**htpasswd Module**

--- # HTPASSWD MODULE EXAMPLE

- hosts: aptserver user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Install the python dependencies apt: pkg=python-passlib state=latest

- name: Adding a user to web site authentication

htpasswd: path=/etc/apache2/.htpasswd name=test2 state=present

**Full Include Tasks**

--- # FULL INCLUDE TASKS EXAMPLE

- hosts: apacheweb user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- include: plays/packages.yml

- name: Verify the telnet package is installed

raw: yum list installed | grep telnet > /home/test/pkgs.log

**Local Action Playbook**

--- # LOCAL ACTION PLAYBOOK

- hosts: 127.0.0.1 connection: local tasks:

- name: Install Telnet Client yum: pkg=telnet state=latest

**Local Action Demo**

--- # LOCALACTION DEMO

- hosts: apacheweb user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Ping application server before we run our install

local\_action: command ping -c 4 tcox5

- name: Install Lynx on remote server yum: pkg=lynx state=latest

**Lookup Playbook**

--- # LOOKUP PLAYBOOK EXAMPLE

- hosts: apacheweb user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- debug: msg=”{{ lookup(‘env’,’HOME’) }} is the value listed”

**Loop Playbook Examples**

--- # LOOP Playbook Example

- hosts: apacheweb user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Add a list of users

user: name={{ item }} state=present

with\_items:

- user1

- user2

- user3

--- # LOOP Playbook Example

- hosts: apacheweb user: test

sudo: sudo connection: ssh gather\_facts: no tasks:

- name: Add a list of users

user: name=user1 state=present

**Mail Module**

--- # MAIL MODULE EXAMPLE

- hosts: aptserver user: test connection: ssh tasks:

- name: Send an email to test user indicating build completion mail:

host=’localhost’ port=25 to=”test”

subject=”Our Host is Finished Deploying”

body=’System called {{ ansible\_hostname }} has been successfully set up’

**modprobe Module**

--- # MODPROBE MODULE EXAMPLE

- hosts: appserver user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Add the dummy module to the remote kerneL

modprobe: name=dummy state=absent

**Mount Module**

--- # MOUNT MODULE EXAMPLE

- hosts: appserver user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: mount the remote data partition

mount: name=/mnt/data src=/dev/xvdf1 fstype=ext3 opts=rw state=present

**My First YAML Playbook**

--- # My First YAML Playbook for Ansible

- hosts: apacheweb user: test

sudo: yes connection: ssh gather\_facts: no vars:

playbook\_version: 0.1b vars\_files:

- conf/copyright.yml

- conf/webdefaults.yml tasks:

- name: Install Apache Web Server

action: yum name=httpd state=installed

- name: Verify the Lynx Web Browser action: yum name=lynx state=present

--- # My First YAML Playbook for Ansible

- hosts: apacheweb user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Install Apache Web Server

action: yum name=httpd state=installed notify: Restart HTTPD

handlers:

- name: Restart HTTPD

action: service name=httpd state=restarted

**mysql\_db Module**

--- # MYSQL\_DB MODULE DEMO

- hosts: appserver user: test

sudo: yes connection: ssh gather\_facts: yes tasks:

- name: Install the Python MySQL Support Libraries yum: pkg=MySQL-python state=latest

- name: Create a New Test DB called MyNewDB

mysql\_db: name=MyNewDB state=present login\_user=root login\_password=password123

**mysql\_user Module**

--- # MYSQL\_USER MODULE DEMO

- hosts: appserver user: test

sudo: yes connection: ssh gather\_facts: yes tasks:

- name: Install the MySQL Python Support Library yum: pkg=MySQL-python state=latest

- name: Create a new user called BOB and give him all access

mysql\_user: name=bob password=123password priv=\*.\*:ALL state=present login\_user=root login\_password=password123

**Package Module**

--- # PACKAGE MODULE EXAMPLE

- hosts: apacheweb user: test

sudo: yes connection: ssh tasks:

- name: Install Apache Web Server

action: package name=telnet state=latest

**Pause Module**

--- # The Pause Module

- hosts: apacheweb sudo: yes gather\_facts: no tasks:

- name: Install HTTPD

action: yum name=httpd state=installed

- name: Pausing pause:

prompt: Press ENTER to Continue...

- name: Verify lynx installation

action: yum name=lynx state=present

**Ping Module**

--- # PING MODULE EXAMPLE

- hosts: all user: test connection: ssh gather\_facts: no tasks:

- name: Ping all the hosts ping:

**Prompt for User Package Example**

--- # PROMPT FOR USER PACKAGE EXAMPLE

- hosts: apacheweb user: test

sudo: yes connection: ssh gather\_facts: no vars:

playbook\_version: 0.01b

vars\_prompt:

- name: pkgtoinstall

prompt: Install Which Package?

default: telnet private: no

tasks:

- name: Install the indicated package

yum: pkg={{ pkgtoinstall }} state=latest

**Raw Module**

--- # RAW MODULE EXAMPLE

- hosts: apacheweb user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Find the system uptime for the ‘hosts’ above raw: /usr/bin/uptime > uptime.log

**run\_once Playbook Example**

--- # RUNONCE PLAYBOOK EXAMPLE

- hosts: all user: test sudo: yes

connection: ssh gather\_facts: no tasks:

- name: Run the uptime command on all hosts and log it raw: /usr/bin/uptime >> /home/test/uptime.log

- name: List the /var directory and log it raw: ls -al /var >> /home/test/dir.list run\_once: true

**Script Module**

--- # SCRIPT MODULE EXAMPLE

- hosts: apacheweb user: test connection: ssh sudo: yes gather\_facts: no tasks:

- script: /home/test/Playbooks/system\_uptime.sh creates=/home/test/uptime.log

$ANSIBLE\_VAULT;1.1;AES25665656664643063623064306233383838316666346138343635

3666643037386265313462656162353130393664643332313332303633393931633964376531

300a623732633765393335666635643066353362396263646530653634636362313262616131

363462353663386338623731316437326663376261623838656666640a663062313561376231

3564323761626630313939396530363233336666316530313361313634303961373864313034

3962363332343162346261303536376362

**SELinux Module**

--- # SELINUX MODULE EXAMPLE

- hosts: apacheweb user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Change SELinux Configuration to Permissive

selinux: policy=targeted state=permissive

**Service Module**

--- # SERVICE MODULE EXAMPLE

- hosts: apacheweb user: test

sudo: yes connection: ssh tasks:

- name: Install Web Server

action: yum name=httpd state=installed

- name: Start the Web Server service: name=httpd state=started

- name: Enable HTTPD After Reboot service: name=httpd enabled=yes

**set\_fact Module**

--- # SET\_FACT MODULE EXAMPLE

- hosts: appserver sudo: yes

user: test connection: ssh gather\_facts: no vars:

playbook\_version: 0.1

tasks:

- name: Local Variable Display set\_fact:

singlefact: SOMETHING

- debug: msg={{ playbook\_version }}

- debug: msg={{ singlefact }}

**Shell Module**

--- # SHELL MODULE EXAMPLE

- hosts: apacheweb user: test

sudo: sudo connection: ssh gather\_facts: no tasks:

- name: Executing a remote command - uptime shell: /usr/bin/uptime >> uptime.log

args:

chdir: logs/

creates: uptime.log

**Start At Playbook Example**

--- # START AT PLAYBOOK EXAMPLE

- hosts: apacheweb user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Install Telnet

yum: pkg=telnet state=latest

- name: Install Lynx

yum: pkg=lynx state=latest

- name: Install at

yum: pkg=at state=latest

**Stat Module**

--- # STAT MODULE EXAMPLE

- hosts: apacheweb user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- stat: path=/home/test/abc register: p

- debug: msg=”The Path Exists and is a Directory”

when: p.stat.isdir is defined and p.stat.isdir

**Tag Functionality**

--- # TAG FUNCTIONALITY EXAMPLE

- hosts: apacheweb user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Install the telnet and lynx packages yum: pkg={{ item }} state=latest with\_items:

- telnet

- lynx tags:

- packages

- name: Verify that telnet was installed

raw: yum list installed | grep telnet > /home/test/pkg.log tags:

- logging

---

- hosts: 127.0.0.1 user: root connection: local gather\_facts: no tasks:

- name: Showing remote status

raw: /usr/bin/uptime > /root/uptime.logt

**Unarchive Module**

--- # UNARCHIVE MODULE EXAMPLE

- hosts: aptserver user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: copy and unarchive a file

unarchive: src=/home/test/local/test.tar.gz dest=/home/test/local copy=no

**Until Example**

--- # UNTIL EXAMPLE

- hosts: apacheweb sudo: yes connection: ssh user: test gather\_facts: no tasks:

- name: Installing Apache Web Server yum: pkg=httpd state=latest

- name: Verify Service Status shell: systemctl status httpd register: result

until: result.stdout.find(“active (running)”) != -1 retries: 5

delay: 5

- debug: var=result

**User Module**

--- # USER MODULE EXAMPLE

- hosts: apacheweb user: test

sudo: yes gather\_facts: no connection: ssh tasks:

- name: Add the user called tstapache to the apache web client

user: name=tst comment=”tst user” shell=/bin/bash groups=wheel append=yes

control\_server: tcox3.mylabserver.com

web\_root: /var/www/html/

**wait\_for Module**

--- # The Wait For Module

- hosts: apacheweb sudo: yes gather\_facts: no tasks:

- name: Installing Apache Tomcat

action: yum name=tomcat state=installed

- name: Waiting for Port 8080 to Listen

wait\_for: port: 8080 state: started

- name: Verifying Lynx Installation

action: yum name=lynx state=present

**When Playbook Example**

--- # WHEN Playbook Example

- hosts: aptserver user: test

sudo: yes connection: ssh vars:

playbook\_type: conditionalexample

vars\_files:

- conf/copyright.yml

- conf/webdefaults.yml tasks:

- name: Install Apache Appropriate to the Distribution Type (Debian/Ubuntu)

command: apt-get -y install apache2 when: ansible\_os\_family == “Debian”

- name: Install Apache Appropriate to the Distribution Type (RedHat/CentOS)

command: yum -y install httpd

when: ansible\_os\_family == “RedHat”

**Yum Module**

--- # Yum Module Example

- hosts: apacheweb user: test

sudo: yes connection: ssh gather\_facts: no tasks:

- name: Equivalent of YUM UPGRADE

action: yum name=\* state=latest