Wiculty Source | Ansible Playbook 01

1)playbook for java installation
[]vi java.yml
```
---
- hosts: webserver
  remote_user: root
  become: yes
  tasks:
  - name: installalling java jdk 8 and it's dependencies
    yum: name=java-1.8.0-openjdk state=installed
```
---
2)Playbook for tomcat
[]vi tomcat.yml
```
---
- hosts: webserver
  remote_user: root
  become: yes
  tasks:
    - name: removing older version of tomcat
              file: path=/usr/local/tomcat state=obsent
            - name: download tomcat
              get_url: url=https://downloads.apache.org/tomcat/tomcat-8/v8.5.58/bin/apache-tomcat-8.5.58.tar.gz
            - name: create tomcat folder
              command: mkdir /usr/local/tomcat
            - name: extract archive
              unarchive: src=/tmp/apache-tomcat-8.0.9.tar.gz dest=/usr/local/tomcat
```
---
3) role for mysql
```
---
- hosts: webserver
  remote_user: root
  become: yes
  gather_facts: false
  roles:
    - mysql
```

[]ansible-galaxy init mysql
initializing the role
[]cd /etc/ansible/rloes
[]ansible-galaxy init mysql
[]cd roles
[]mysql
[]ls -la

[]cd defaults
[]vi main.yml
```
mysql_user: mysql
mysql_port: 3306
mysql_datadir: /var/lib/mysql
mysql_bind: 0.0.0.0
mysql_pkg: mysql_server
mysql_pid: /var/run/mysql/mysqld.pid
mysql_socket: /var/lib/mysqld/mysql.sock
mysql_confpath: /etc/mycnf
mysql_service: mysql
```

```
[]cd handlers.yml
[]vi main.yml
---
- name: restart mysql service
  service: name="{{mysql_service}}" state=restart

[]cd tasks/
[]vi main.yml
---
- name: install mysql servername
  yum: name="{{mysql-pkg}}" state=present
- name: create mysql config
  template: src="{{mycnf.j2}}" dest="{{mysql_cnfpath}}"
  notify:
   - restart: mysql service

[]cd templates
[]vi mycnf.j2
user="{{mysql_user}}"
pid_file="{{mysql_pid}}"
socket="{{mysql_socket}"
port="{{mysql=port}}"
datadir="{{mysql_datadir}}"
bind_address="{{mysql_bind}}"
```

---

4)Playbook for Jenkins:

```
---
- hosts: webserver
  remote_user: root
  become: yes
  gather_facts: false
  tasks:
    - name: Ensure Jenkins Repository is Installed
      yum_repository:
        name: jenkins
        state: present
        description: Official Jenkins Yum Repo
        baseurl: http://pkg.jenkins.io/redhat
            - name: installing jenkins
      yum :
        name: jenkins
        update_cache: yes
        state: present
    - name: Enable and Start the Jenkins Service
      service:
        name: jenkins
        enabled: yes
        state: started
    - name: Open Firewall Port
      firewalld:
        zone: public
        port: 8080/tcp
        permanent: true
        state: enabled
        immediate: true
```

types or states:

stop

started
enabel
restart

./startup.sh
[]./stratup.sh
[]./shutdown.sh

Note:
update_cache=yes tells Ansible's +module to refresh the caches before applying whatever change is necessary (if any)
When immediate is set to true on an Command component, the action is invoked in the Apply Request Values phase
---
7)Playbook for ping test

```
- hosts: webserver
  remote_user: root
  become: yes
  tasks:
    - name: ping machine
      ping:
```
---
5)Installing Nexus through playbook

```
- hosts: webserver
  remote_user: root
  become: yes
  tasks:
   - name: Download Nexus package
     get_url: url="http://download.sonatype.com/nexus/oss/{{ nexus_package }}" dest=/root
```
---
6)installing python

```
- hosts: webserver
  remote_user: root
  become: yes
  - name: Install Python
    yum: name=python state=installed
    raw: yum -y install python
    command: yum -y install python
```


```
- hosts: webserver
  remote_user: root
  become: yes
  tasks:
  - name: Add MongoDB repository
    apt_repository: repo='deb http://downloads-distro.mongodb.org/repo/ubuntu-upstart dist 10gen'
state=present
  - name: install mongodb
    apt: pkg=mongodb-org state=latest update_cache=yes
    notify:
    - start mongodb
  handlers:
    - name: start mongodb
      service: name=mongod state=restarted
```
---
7)Playbook to install nginx

```
- hosts: all
  tasks:
    - name: ensure nginx is at the latest version
      yum: name=nginx state=installed
    - name: start nginx
      service:
          name: nginx
          state: started
```

---

8)Playbook for selinum

```
---
- hosts: webserver
  remote_user: root
  become: yes
  ---
- name: Download the gecko driver
  get_url: url="https://github.com/mozilla/geckodriver/releases/download/v{{ gecko_version }}/geckodriver-v{{ gecko_version }}-macos.tar.gz" dest=/tmp mode=0777

- name: Install the gecko driver
  unarchive: src="/tmp/geckodriver-v{{ gecko_version }}-macos.tar.gz" dest=/usr/local/bin/ mode=0700 copy=no

- name: Download the chrome driver
  get_url: url="http://chromedriver.storage.googleapis.com/{{ chrome_version }}/chromedriver_mac32.zip" dest=/tmp mode=0777

- name: Install the chrome driver
  unarchive: src=/tmp/chromedriver_mac32.zip dest=/usr/local/bin/ mode=0700 copy=no

- name: Download the safari extension
  get_url: url="http://selenium-release.storage.googleapis.com/{{ safari_version }}/SafariDriver.safariextz" dest=/tmp mode=0777

- name: Open the safari extension
  command: open /tmp/SafariDriver.safariextz

- name: Download selenium zip
  get_url: url="https://github.com/SeleniumHQ/selenium/archive/selenium-{{ selenium_version }}.zip" dest=/tmp mode=0777

- name: Unzip selenium
  unarchive: src="/tmp/selenium-selenium-{{ selenium_version }}.zip" dest=/tmp mode=0700 copy=no

- name: Build Selenium
  command: ./go clean release
  args:
    chdir: "/tmp/selenium-selenium-{{ selenium_version }}/"
    creates: "/tmp/selenium-selenium-{{ selenium_version }}/build/dist/selenium-server-standalone-{{ selenium_version }}.jar"

- name: Move Selenium build
  command: mv "selenium-server-standalone-{{ selenium_version }}.jar" /usr/local/bin/selenium-server.jar
  args:
    chdir: "/tmp/selenium-selenium-{{ selenium_version }}/build/dist/"
    creates: /usr/local/bin/selenium-server.jar

- name: Create script for selenium
  template: src=selenium dest=/usr/local/bin/selenium mode=0700
```
