|  |  |
| --- | --- |
| **DOCUMENT RULES:** | |
| **Task Number / Name:** | **Ansible** |
| **Task name & column name should be written:** | **Bold (CTRL+B)** |
| **Commands should be written in the after # sign:** | *Italic (CTRL+I) #hostname* |
| **Output photo should be cropped or compressed:**  **Photo could be more than one:**  **If you need extra lines, add the line next after it:** | ***Description photo should be with title bar (CTRL + I + B)*** |
| **All other text should be written:** | Standard |
| **Font name and text size:** | Calibri and 9 |
| **Group name:** | Dev\_ops\_ |
| **Student name and surname:** | Murad Abbaszade |
| **E-mail:** | [muradabbaszade6@gmail.com](mailto:muradabbaszade6@gmail.com) |
| **WhatsApp number:** | **+994703664205** |

|  |  |
| --- | --- |
| 1.Ansible(101) |  |
| apt install -y python3-pip && mkdir -p /opt/kata-materials && cd /opt/kata-materials && git clone --depth 1 <https://github.com/irixjp/katacoda-scenarios.git> . && pip install virtualenv && virtualenv ansible && source /opt/kata-materials/ansible/bin/activate && cd tools && bash ./kata\_setup.sh && cd ~/ |  |
| In this exercise, we will use an environment built as follows. Three servers, node-1 , node-2 , and node-3 , are running, and we will use Ansible to perform various automated operations on them.  First, please run the following command. This is using Ansible to check the disk usage of the three exercise nodes.  cd ~/  ansible all -m shell -a 'df -h' |  |
| ansible --version |  |
| If you run the ansible command with the --version option, it will output some basic information about the execution environment. This includes the version and the Python version you are using. Here we will focus on the following line.   * config file = /root/.ansible.cfg   cat ~/.ansible.cfg |  |
| cat ~/inventory\_file |  |
|  |  |
| ansible web -m ping -o |  |
| ansible web -i ~/inventory\_file -m ping -o  ansible node-1 -m ping -o |  |
| ansible node-1,node-3 -m ping -o |  |
| ansible all -m ping -o |  |
| ansible web --list-hosts |  |
| ansible node-2 --list-hosts |  |
| ansible all -u centos --private-key ~/aitac-automation-keypair.pem -m ping |  |
| ansible-doc -l |  |
| ansible-doc dnf |  |
| ansible all -m ping |  |
| ansible all -m shell -a 'hostname' |  |
| ansible all -m shell -a 'uname -a' |  |
| ansible all -m shell -a 'date' |  |
| ansible all -m shell -a 'df -h' |  |
| ansible all -m shell -a 'rpm -qa |grep bash' |  |
| ansible node-1 -b -m dnf -a 'name=screen state=latest' |  |
| ansible node-1 -m shell -a 'which screen' |  |
| Ansible(102) |  |
| apt install -y python3-pip && mkdir -p /opt/kata-materials && cd /opt/kata-materials && git clone --depth 1 <https://github.com/irixjp/katacoda-scenarios.git> . && pip install virtualenv && virtualenv ansible && source /opt/kata-materials/ansible/bin/activate && cd tools && bash ./kata\_setup.sh && cd ~/ |  |
| Please edit ~/working/vars\_debug\_playbook.yml as follows.  ---  - hosts: node-1  gather\_facts: no  tasks:  - name: print all variables  debug:  var: vars  - name: get one variable  debug:  msg: "This value is {{ vars.ansible\_version.full }}"  cd ~/working  ansible-playbook vars\_debug\_playbook.yml |  |
| Please edit ~/working/vars\_play\_playbook.yml as follows.  ---  - hosts: node-1  gather\_facts: no  vars:  play\_vars:  - order: 1st word  value: ansible  - order: 2nd word  value: is  - order: 3rd word  value: fine  tasks:  - name: print play\_vars  debug:  var: play\_vars  - name: access to the array  debug:  msg: "{{ play\_vars[1].order }}"  - name: join variables  debug:  msg: "{{ play\_vars[0].value}} {{ play\_vars[1].value }} {{ play\_vars[2].value  }}"  cd ~/working  ansible-playbook vars\_play\_playbook.yml |  |
| Let's edit ~/working/vars\_task\_playbook.yml as follows.  ---  - hosts: node-1  gather\_facts: no  vars:  task\_vars: 100  tasks:  - name: print task\_vars 1  debug:  var: task\_vars  - name: override task\_vars  debug:  var: task\_vars  vars:  task\_vars: 20  - name: print task\_vars 2  debug:  var: task\_vars  ansible-playbook vars\_task\_playbook.yml |  |
| ansible-playbook vars\_task\_playbook.yml -e 'task\_vars=50' |  |
| Let's create a playbook that uses these variables.  ---  - hosts: all  gather\_facts: no  tasks:  - name: print group\_vars  debug:  var: vars\_by\_group\_vars  - name: print host vars  debug:  var: vars\_by\_host\_vars  - name: vars\_by\_group\_vars + vars\_by\_host\_vars  set\_fact:  cal\_result: "{{ vars\_by\_group\_vars + vars\_by\_host\_vars }}"  - name: print cal\_vars  debug:  var: cal\_result  Edit ~/working/loop\_playbook.yml as follows.  ---  - name: add users by loop  hosts: node-1  become: yes  vars:  user\_list:  - apple  - orange  - pineapple  tasks:  - name: add a user  user:  name: "{{ item }}"  state: present  loop: "{{ user\_list }}"  cd ~/working  ansible-playbook loop\_playbook.yml |  |
| ansible node-1 -b -m shell -a 'cat /etc/passwd' |  |
| ansible-playbook loop\_playbook.yml |  |
| Let's write the following ~/working/when\_playbook.yml .  ---  - name: start httpd if it's stopped  hosts: node-1  become: yes  tasks:  - name: install httpd  yum:  name: httpd  state: latest  - name: check httpd processes is running  shell: ps -ef |grep http[d]  register: ret  ignore\_errors: yes  changed\_when: no  - name: print return value  debug:  var: ret  - name: start httpd (httpd is stopped)  service:  name: httpd  state: started  when:  ansible-playbook when\_playbook.yml |  |
| ansible-playbook when\_playbook.yml |  |
| ansible node-1 -m fetch -a 'src=/etc/httpd/conf/httpd.conf dest=files/httpd.conf flat=yes' |  |
| Edit ~/working/handler\_playbook.yml as follows:  ---  - name: restart httpd if httpd.conf is changed  hosts: node-1  become: yes  tasks:  - name: Copy Apache configuration file  copy:  src: files/httpd.conf  dest: /etc/httpd/conf/  notify:  - restart\_apache  handlers:  - name: restart\_apache  service:  name: httpd  state: restarted |  |
| Edit the ~/working/block\_playbook.yml  ---  - name: using block statement  hosts: node-1  become: yes  tasks:  - name: Install, configure, and start Apache  block:  - name: install httpd  yum:  name: httpd  state: latest  - name: start & enabled httpd  service:  name: httpd  state: started  enabled: yes  - name: copy index.html  copy:  src: files/index.html  dest: /var/www/html/  when:  ansible-playbook block\_playbook.yml -e 'exec\_block=no' |  |
| ansible-playbook block\_playbook.yml -e 'exec\_block=yes' |  |
| Create ~/working/rescue\_playbook.yml as follows.  ---  - name: using block, rescue, always statement  hosts: node-1  become: yes  tasks:  - block:  - name: block task  debug:  msg: "message from block"  - name: check error flag in block  assert:  that:  - error\_flag == 'no'  rescue:  - name: rescue task  debug:  msg: "message from rescue"  - name: check error flag in rescue  assert:  that:  - error\_flag == 'no'  always:  - name: always task  debug:  ansible-playbook rescue\_playbook.yml -e 'error\_flag=no' |  |
| ~/working/template\_playbook.yml and try to run the template. Please edit template\_playbook.yml as follows.  ---  - name: using template  hosts: web  become: yes  tasks:  - name: install httpd  yum:  name: httpd  state: latest  - name: start & enabled httpd  service:  name: httpd  state: started  enabled: yes  - name: Put index.html from template  template:  src: templates/index.html.j2  dest: /var/www/html/index.html  cd ~/working  ansible-playbook template\_playbook.yml -e 'LANG=JP' |  |
| ansible-playbook rescue\_playbook.yml -e 'error\_flag=yes' |  |
| ansible web -m uri -a 'url=http://localhost/ return\_content=yes' |  |
| ansible-playbook template\_playbook.yml -e 'LANG=EN' |  |
| ansible web -m uri -a 'url=http://localhost/ return\_content=yes' |  |
| ansible node-1 -m debug -a 'msg={{ hoge | default("abc") }}' |  |
| ansible node-1 -e 'str=abc' -m debug -a 'msg="{{ str | upper }}"' |  |
| ansible node-1 -m debug -a 'msg="{{ [5, 1, 10] | min }}"' |  |
| ansible node-1 -m debug -a 'msg="{{ [5, 1, 10] | max }}"' |  |
| Ansible(103) |  |
| apt install -y python3-pip && mkdir -p /opt/kata-materials && cd /opt/kata-materials && git clone --depth 1 <https://github.com/irixjp/katacoda-scenarios.git> . && pip install virtualenv && virtualenv ansible && source /opt/kata-materials/ansible/bin/activate && cd tools && bash ./kata\_setup.sh && cd ~/ |  |
| Edit ~/working/roles/web\_setup/templates/index.html.j2  <html><body>  <h1>This server is running on {{ inventory\_hostname }}.</h1>  {% if LANG == "JP" %}  Konnichiwa!  {% else %}  Hello!  {% endif %}  </body></html>  cd ~/working/roles/web\_setup  ansible node-1 -b -m yum -a 'name=httpd state=latest'  ansible node-1 -m fetch -a 'src=/etc/httpd/conf/httpd.conf dest=files/httpd.conf flat=yes'  Verify that the file has been retrieved.  ls -l files/ |  |
| Edit ~/working/role\_playbook.yml  Create a playbook that actually calls the role.  ---  - name: using role  hosts: web  become: yes  tasks:  - import\_role:  name: web\_setup  Next, create a ~/working/galaxy\_playbook.yml that uses this role.  ---  - name: using galaxy  hosts: node-1  tasks:  - import\_role:  name: irixjp.role\_example\_hello  - import\_role:  name: irixjp.role\_example\_uptime  ansible-galaxy collection list |  |
| ansible-galaxy collection install cisco.ios  ansible-galaxy collection list |  |
| ansible-doc -l | wc -l  ls -alF ~/.ansible/collections/ansible\_collections/ |  |
| rm -rf ~/.ansible/collections/ansible\_collections/{ansible,cisco}  ls -alF ~/.ansible/collections/ansible\_collections/ |  |
| ansible-galaxy collection list |  |
| ansible-doc -l | wc –l  ansible-galaxy collection install cisco.ios:2.3.1 `  ansible-galaxy collection list |  |
| Please edit ~/working/collections/requirements.yml as follows.  ---  collections:  - name: irixjp.sample\_collection\_hello  version: 1.0.0  cd ~/working  ansible-galaxy collection install -r collections/requirements.yml  ansible-galaxy collection list |  |
| Ansible(104) |  |
| apt install -y python3-pip && mkdir -p /opt/kata-materials && cd /opt/kata-materials && git clone --depth 1 <https://github.com/irixjp/katacoda-scenarios.git> . && pip install virtualenv && virtualenv ansible && source /opt/kata-materials/ansible/bin/activate && cd tools && bash ./kata\_setup.sh && cd |  |
| cd ~/working  ansible-playbook lint\_ok\_playbook.yml  ansible-playbook lint\_ng\_playbook.yml |  |
| ansible-lint lint\_ok\_playbook.yml |  |
| ansible-lint lint\_ng\_playbook.yml |  |
| ansible-lint -L  ansible-lint –T  ansible node-1 -b -m yum -a 'name=httpd state=present' |  |
| ansible node-1 -b -m systemd -a 'name=httpd state=started enabled=yes' |  |
| Edit the file ~/working/testing\_assert\_playbook.yml as follows.  ---  - name: Test with assert  hosts: node-1  become: yes  gather\_facts: no  tasks:  - ignore\_errors: yes  block:  - name: Is httpd package installed?  shell: yum list installed | grep -e '^httpd\.'  register: ret\_httpd\_pkg  - name: check httpd processes is running  shell: ps -ef |grep http[d]  register: ret\_httpd\_proc  - name: Is httpd service enabled?  shell: systemctl is-enabled httpd  register: ret\_httpd\_enabled  - block:  - name: Assert results  assert:  that:  - ret\_httpd\_pkg.rc == 0  - ret\_httpd\_proc.rc == 0  - ret\_httpd\_enabled.rc == 0  cd ~/working  ansible-playbook testing\_assert\_playbook.yml |  |
| ansible node-1 -b -m systemd -a 'name=httpd state=stopped enabled=yes'  ansible-playbook testing\_assert\_playbook.yml  ansible node-1 -b -m systemd -a 'name=httpd state=started enabled=yes'  ansible-playbook testing\_assert\_playbook.yml |  |
| Create the file ~/working/reporting\_playbook.yml as following.  ---  - name: Report with Ansible  hosts: web  gather\_facts: true  tasks:  - name: build report  copy:  content: |  # Server Configuration Reports: {{ inventory\_hostname }}  ---  | name | value |  | ---- | ------ |  {% for key, value in ansible\_default\_ipv4.items() %}  | {{ key }} | {{ value }} |  {% endfor %}  dest: /tmp/setting\_report\_{{ inventory\_hostname }}.md  delegate\_to: localhost    - name: concatenate reports  assemble:  src: /tmp  regexp: 'setting\\_report\\_\*'  dest: setting\_report.md  delimiter: "\n"  run\_once: true  delegate\_to: localhost  cd ~/working  ansible-playbook reporting\_playbook.yml |  |