EPAM Systems, RD Dep.

MTN.*NIX.11 Automated Environment Configuration Management

Ansible. 2

REVISION HISTORY					
Ver.	**Description of Change**	**Author**	**Date**	**Approved**	
				Name	**Effective Date**
<1.0>	Initial revision	Siarhei Beliakou	17-Mar-2017		

Lab Work Task. Web Server Provisioning

Review

Using Ansible v2.2.1 for provisioning nginx + tomcat application stack.

Learning by doing.

Task

On Host Node (Control Machine):

1. Create folder ~/cm/ansible/day-2. All working files are supposed to be placed right there.
2. Spin up clear CentOS6 VM using vagrant (repo with vagrantfile). Verify connectivity to the host using ssh keys (user: vagrant)
3. Create ansible inventory file (name: **inventory**) with remote host connection details:
   - Remote VM hostname/ip/port
   - Remote ssh log in username
   - Connection type

inventory

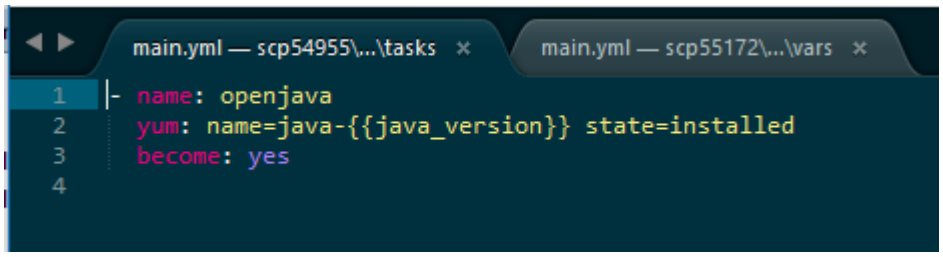
```
[webservers]
websvr
[webservers:vars]
ansible_host=localhost
ansible_port=2222
ansible_connection=ssh
ansible_user=vagrant
ansible_ssh_private_key_file=.vagrant/machines/default/virtualbox/private_key

[local_cm]
localhost_cm

[local_cm:vars]
ansible_connection=local
```

4. Develop a playbook (name: **site.yml**) which is supposed to run against any host (specified in inventory)

   4.1 Develop roles:

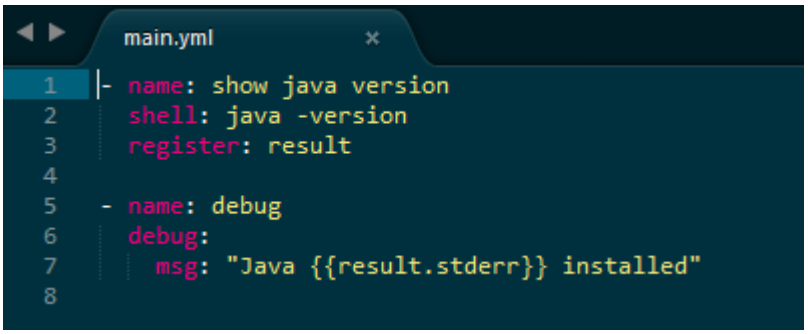
   - **java** (installs java)

./roles/java/tasks/main.yml



   - **java_test** (does only checks that java installed and running properly)

./roles/java_test/tasks/main.yml



- **tomcat** (installs tomcat)

./roles/tomcat/tasks/main.yml

main.yml — scp01807\...\tasks × main.yml — scp03619\...\vars ×

```yaml
- name: group "tomcat_as_group"
  group: name={{tomcat_group}}
  become: yes

- name: user "tomcat_as"
  user: name='{{tomcat_user}}' group='{{tomcat_group}}' home='{{tomcat_home}}'
  become: yes

- name: get tomcat ver. {{tomcat_version}}
  unarchive:
    src: '{{tomcat_download_uri}}'
    dest: '{{tomcat_home}}'
    remote_src: True
    creates: '{{tomcat_home}}/{{tomcat_unarchived}}'
  become: yes
  become_user: '{{tomcat_user}}'

- name: rename tomcat dir
  command: mv {{tomcat_home}}/{{tomcat_unarchived}} {{tomcat_home}}/{{tomcat_version}} creates={{tomcat_home}}/{{tomcat_version}}
  become: yes
  become_user: '{{tomcat_user}}'

- name: startup script
  template:
    src: tomcat.service.j2
    dest: /etc/rc.d/init.d/tomcat
    mode : a+x
  become: yes

- name: tomcat {{tomcat_version}} started
  service: name=tomcat state=started
  become: yes
  become_user: '{{tomcat_user}}'

- shell:
    res=`/sbin/service tomcat status`
    if [[ $res == 'Tomcat is running' ]]; then echo STARTED; fi
  register: result
- name: debug
  debug:
    msg: "Tomcat status: {{result.stdout}}"
```

- **tomcat_test** (does only checks that tomcat installed and running properly)

./roles/tomcat_test/tasks/main.yml

main.yml

```yaml
- name: sheck service status
  shell: service tomcat status
  become: yes
  become_user: root

- name: check tomcat service
  shell: ps aux | grep tomcat
```

- **nginx** (installs nginx)

./roles/nginx/tasks/main.yml

main.yml — scp57454\...\tasks | main.yml — scp00269\...\tasks | vhosts.j2

```yaml
- name: nginx
  yum: name=nginx state=installed
  become: yes

- name: config nginx
  template: src=vhosts.j2 dest=/etc/nginx/conf.d/vhosts.conf
  become: yes
  notify: restart nginx

- name: run nginx
  service: name=nginx state=started
  become: yes

```

./roles/nginx/templates/vhosts.j2

main.yml — scp57454\...\tasks | main.yml — scp00269\...\tasks | vhosts.j2

```
server {
    listen 80;
    server_name 192.168.33.10;
  location / {
      proxy_pass http://127.0.0.1:8080;
    }
}

```

- **nginx_test** (does only checks that nginx installed and running properly)

./roles/nginx_test/tasks/main.yml

4.2 Playbook should consist of **2** Plays:

- Installation
- Verification

inventory | untitled | site.yml

```
- name: install java tomcat nginx
  hosts: webservers
  roles:
  - java
  - tomcat
  - nginx

- name: check java tomcat nginx
  hosts: webservers
  roles:
  - java_test
  - tomcat_test
  - nginx_test

- name: is available from local_cm
  hosts: local_cm
  tasks:
  - name: check tomcat availability from host
    shell: |
     curl -sL -w %{http_code} http://192.168.33.10 -o /dev/null
     if [ $? -eq 0 ]; then
      echo '. curl Success'
     else
      echo '. curl Fail'
     fi
    register: result

  - name: debug
    debug:
      msg: "Curl status: {{result.stdout}}"

```

4.3 Use **handlers** to manage tomcat/nginx configuration changes

./roles/nginx/handlers/main.yml

main.yml

```
- name: restart nginx
  service:
    name: nginx
    state: restarted
  become: yes

```

4.4 Use module **debug** to check configuration during the installation

4.5 Define play/roles variables (at least):

- **tomcat_version**
- **tomcat_home**
- **tomcat_user**
- **tomcat_group**

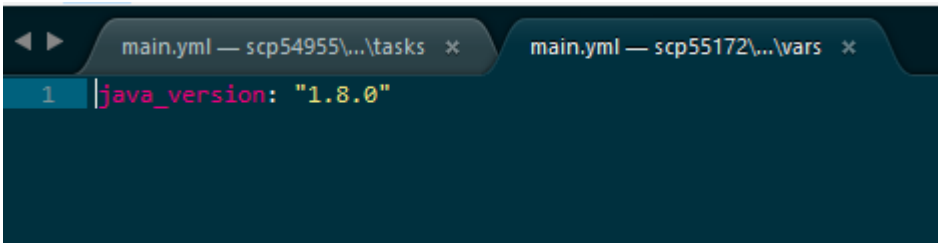
./roles/tomcat/vars/main.yml

main.yml — scp01807\...\tasks × main.yml — scp03619\...\vars ×

```
tomcat_version: '8.5.12'
tomcat_home: '/opt/tomcat'
tomcat_user: tomcat_as
tomcat_group: tomcat_as_group
tomcat_unarchived: 'apache-tomcat-{{tomcat_version}}'

tomcat_download_uri: 'http://archive.apache.org/dist/tomcat/tomcat-{{tomcat_version[:1]}}/v{{tomcat_version}}/bin/apache-tomcat-{{tomcat_version}}.tar.gz
```

- **java_version**

./roles/java/vars/main.yml



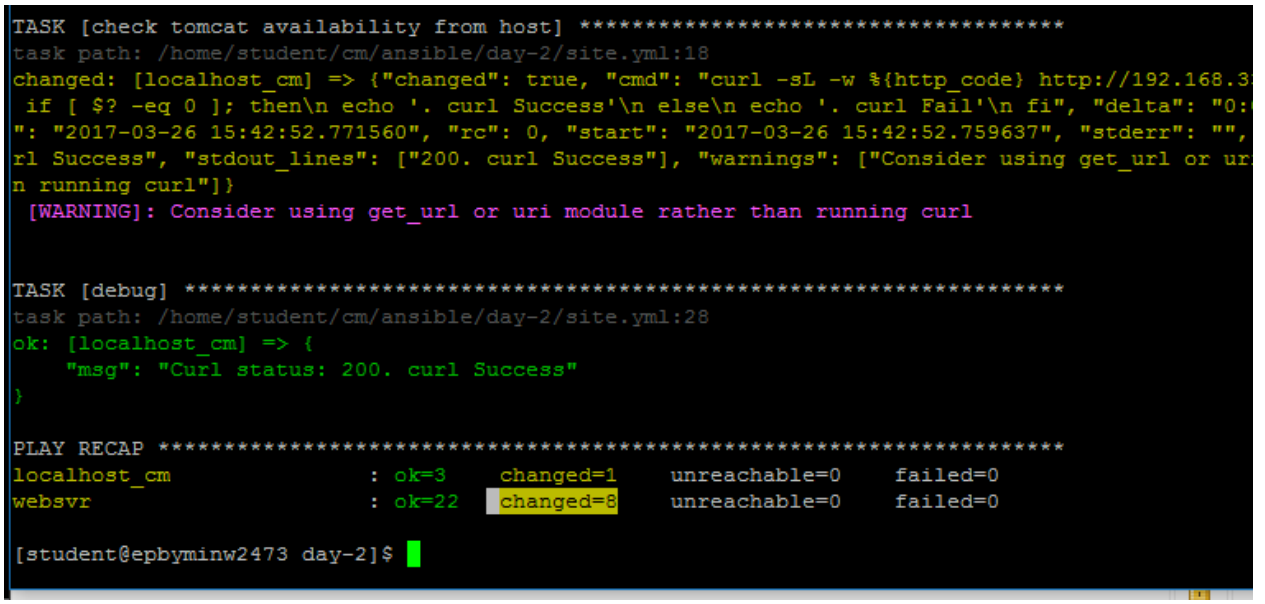
4.6 Every task/handler should have a name section with details of task purpose.

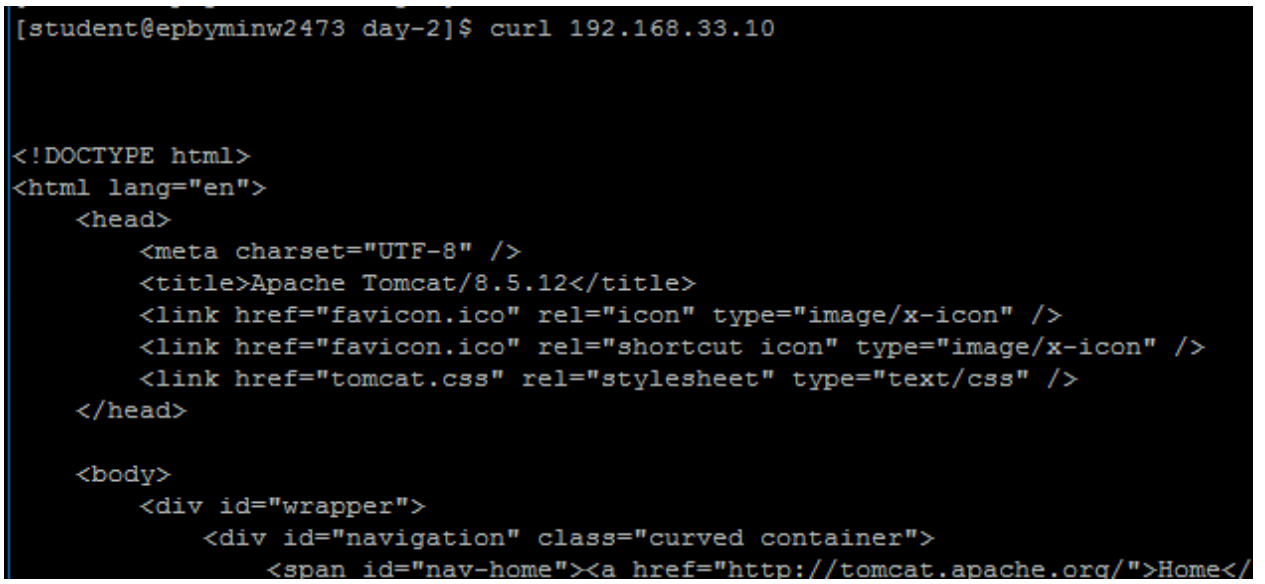
5. Software installation requirements:
   - Tomcat AS should be installed from sources (tar.gz) – download from the official site (http://archive.apache.org/dist/tomcat/).
   - Tomcat AS should be owned (and run) by user specified in variable (default: tomcat_as:tomcat_as_group).
   - Tomcat AS version should be 7.x, 8.x (at least 5 versions), exact version to be installed is taken from appropriate variable.
   - Tomcat installation folder (CATALINA_HOME) is /opt/tomcat/**$version**, where **$version** is the version of tomcat defined in playbook.
   - Java can be installed from CentOS Repositories
   - Use module **yum** to install Nginx
   - Use module **template** for management of nginx cofigs

- Tomcat home page should be available on port 80 (accessible from Control Machile) via nginx.

```
[student@epbyminw2473 day-2]$ ansible-playbook site.yml -i inventory -vv
No config file found; using defaults

PLAYBOOK: site.yml *************************************************************
3 plays in site.yml

PLAY [install java tomcat nginx] ***********************************************

TASK [setup] *******************************************************************
ok: [websvr]

TASK [java : openjava] *********************************************************
task path: /home/student/cm/ansible/day-2/roles/java/tasks/main.yml:1
changed: [websvr] => {"changed": true, "msg": "", "rc": 0, "results": ["Loaded plugi
 Install Process\nLoading mirror speeds from cached hostfile\n * base: ftp.byfly.by\
y\n * extras: ftp.byfly.by\n * updates: ftp.byfly.by\nResolving Dependencies\n--> Ru
 Package java-1.8.0-openjdk.x86_64 1:1.8.0.121-0.b13.el6_8 will be installed\n--> Pr
```

```
TASK [setup] *******************************************************************
ok: [localhost_cm]

TASK [check tomcat availability from host] **************************************
task path: /home/student/cm/ansible/day-2/site.yml:18
changed: [localhost_cm] => {"changed": true, "cmd": "curl -sL -w %{http_code} 
 if [ $? -eq 0 ]; then\n echo '. curl Success'\n else\n echo '. curl Fail'\n fi
": "2017-03-26 15:41:43.589097", "rc": 0, "start": "2017-03-26 15:41:42.855424
rl Success", "stdout_lines": ["200. curl Success"], "warnings": ["Consider usi
n running curl"]}
 [WARNING]: Consider using get_url or uri module rather than running curl


TASK [debug] *******************************************************************
task path: /home/student/cm/ansible/day-2/site.yml:28
ok: [localhost_cm] => {
    "msg": "Curl status: 200. curl Success"
}

PLAY RECAP *********************************************************************
localhost_cm               : ok=3    changed=1    unreachable=0    failed=0
websvr                     : ok=23   changed=18   unreachable=0    failed=0

[student@epbyminw2473 day-2]$
```





6. Verification Procedure: playbook will be checked by instructor's CI system as follows:
   6.1 Connect to student's host by ssh (username "student") with own ssh key.
   6.2 Go into the folder mentioned in point 1
   6.3 Destroy/Launch VM: vagrant destroy && vagrant up
   6.4 Execute VM provisioning: ls ansible-playbook site.yml -i inventory -vv
   6.5 If previous steps are done successfully, instructor will check report (pdf-file)
7. Feedback: report issues/problems you had during the development of playbook and time spent for development.