

EPAM Systems, RD Dep., 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

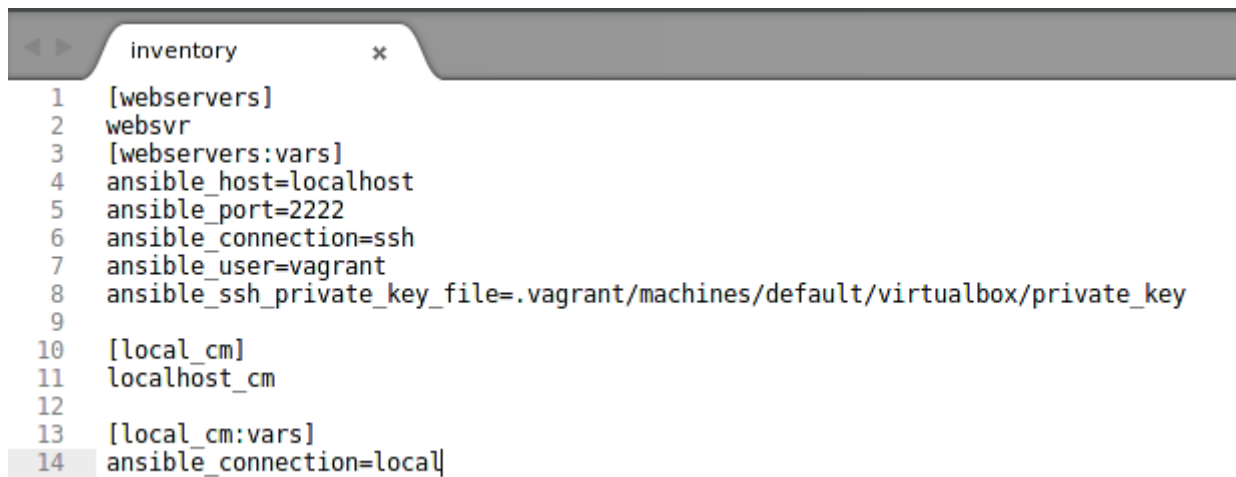
Using Ansible v2.2.1 for provisioning nginx + tomcat application stack.

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



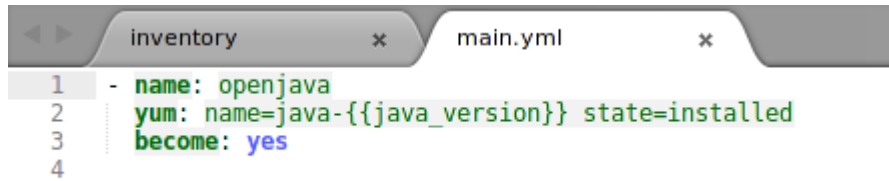
```
1 [webservers]
2 websvr
3 [webservers:vars]
4 ansible_host=localhost
5 ansible_port=2222
6 ansible_connection=ssh
7 ansible_user=vagrant
8 ansible_ssh_private_key_file=.vagrant/machines/default/virtualbox/private_key
9
10 [local_cm]
11 localhost_cm
12
13 [local_cm:vars]
14 ansible_connection=local
```

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)

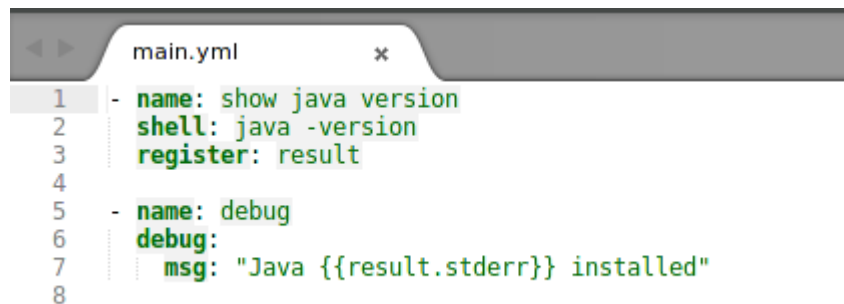
`./roles/java/tasks/main.yml`



```
1 - name: openjava
2   yum: name=java-{{java_version}} state=installed
3   become: yes
4
```

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

`./roles/java_test/tasks/main.yml`



```
1 - name: show java version
2   shell: java -version
3   register: result
4
5 - name: debug
6   debug:
7     msg: "Java {{result.stderr}} installed"
8
```

- **tomcat** (installs tomcat)

./roles/tomcat/tasks/main.yml

```
main.yml
1 - name: group "tomcat_as_group"
2   group: name={{tomcat_group}}
3   become: yes
4
5 - name: user "tomcat_as"
6   user: name={{tomcat_user}} group={{tomcat_group}} home={{tomcat_home}}
7   become: yes
8
9 - name: get tomcat ver. {{tomcat_version}}
10  unarchive:
11    src: '{{tomcat_download_uri}}'
12    dest: '{{tomcat_home}}'
13    remote_src: True
14    creates: '{{tomcat_home}}/{{tomcat_unarchived}}'
15    become: yes
16    become_user: '{{tomcat_user}}'
17
18 - name: rename tomcat dir
19   command: mv '{{tomcat_home}}/{{tomcat_unarchived}}' '{{tomcat_home}}/{{tomcat_version}}' creates={{tomcat_home}}/{{tomcat_version}}
20   become: yes
21   become_user: '{{tomcat_user}}'
22
23 - name: startup script
24   template:
25     src: tomcat.service.j2
26     dest: /etc/rc.d/init.d/tomcat
27     mode: a+x
28     become: yes
29
30 - name: tomcat {{tomcat_version}} started
31   service: name=tomcat state=started
32   become: yes
33   become_user: '{{tomcat_user}}'
34
35 - shell: |
36   res=`/sbin/service tomcat status`
37   if [[ $res == 'Tomcat is running' ]]; then echo STARTED; fi
38   register: result
39 - name: debug
40   debug:
41     msg: "Tomcat status: {{result.stdout}}"
```

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

`./roles/tomcat_test/tasks/main.yml`

```
main.yml x
1 - name: sheck service status
2   shell: service tomcat status
3   become: yes
4   become_user: root
5
6 - name: check tomcat service
7   shell: ps aux | grep tomcat
```

- **nginx** (installs nginx)

`./roles/nginx/tasks/main.yml`

```
main.yml x
1 |- name: nginx
2   yum: name=nginx state=installed
3   become: yes
4
5 - name: config nginx
6   template: src=vhosts.j2 dest=/etc/nginx/conf.d/vhosts.conf
7   become: yes
8   notify: restart nginx
9
10 - name: run nginx
11   service: name=nginx state=started
12   become: yes
13
```

`./roles/nginx/templates/vhosts.j2`

```
vhosts.j2 x
1 server {
2     listen 80;
3     server_name 192.168.33.10;
4     location / {
5         proxy_pass http://127.0.0.1:8080;
6     }
7 }
8
```

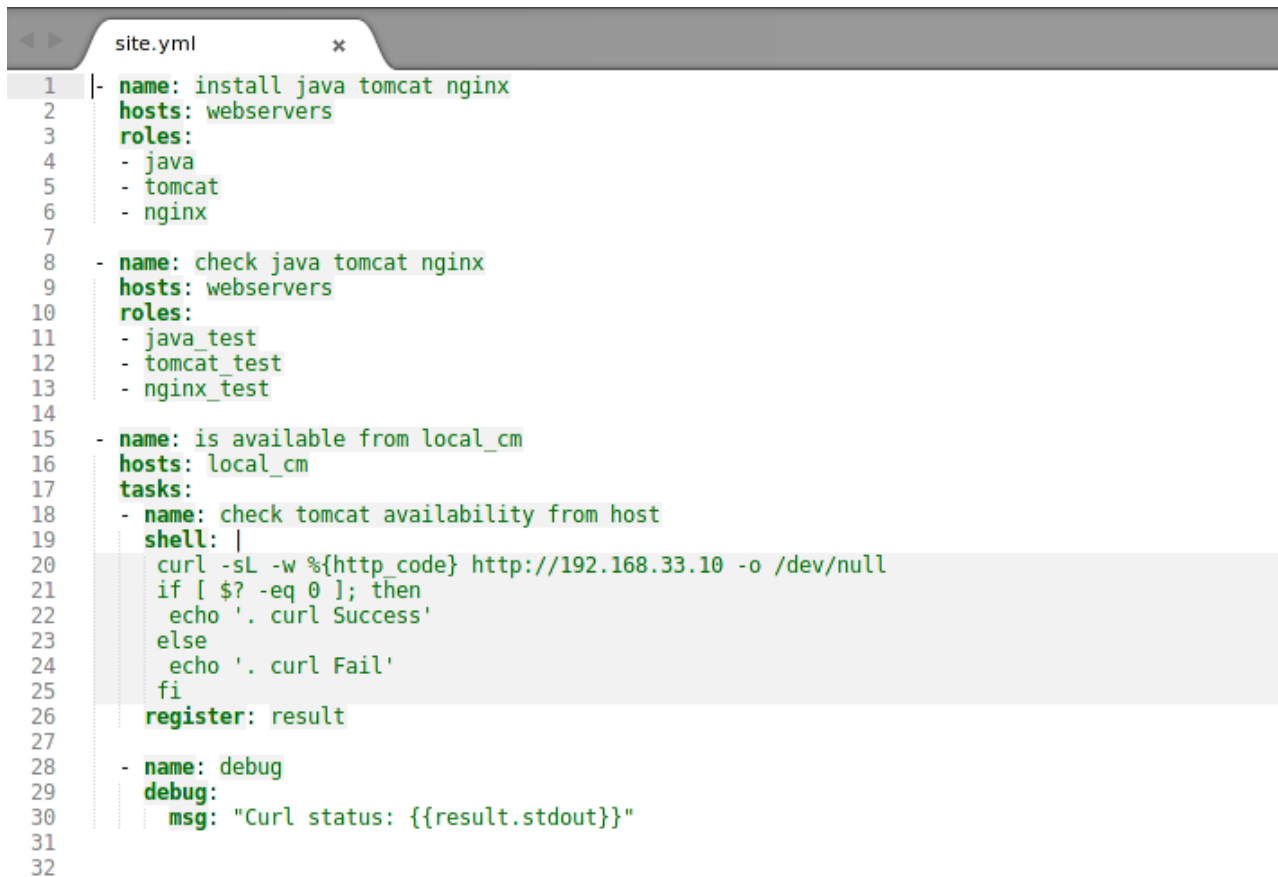
- **nginx_test** (does only checks that nginx installed and running properly)

`./roles/nginx_test/tasks/main.yml`

```
main.yml
1 - name: check nginx is running
2   shell: service nginx status
3   become: yes
4
5 - shell: |
6     res=`/sbin/service nginx status | grep 'is running'`
7     if [[ $res != '' ]]; then echo STARTED; fi
8   register: result
9
10 - name: debug
11   debug:
12     msg: "nginx status: {{result.stdout}}"
13
14 - name: get start time
15   shell: |
16     echo Now: `date`
17     pid=`ps aux | grep nginx | grep -v grep | head -1 | awk '{print $2}'`
18     stat -c %z /proc/$pid
19   become: yes
20
```

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


- Installation
- Verification



```
1 1|- name: install java tomcat nginx
2   hosts: webservers
3   roles:
4     - java
5     - tomcat
6     - nginx
7
8 2- name: check java tomcat nginx
9   hosts: webservers
10  roles:
11    - java_test
12    - tomcat_test
13    - nginx_test
14
15 3- name: is available from local_cm
16   hosts: local_cm
17   tasks:
18     - name: check tomcat availability from host
19       shell: |
20         curl -sL -w %{http_code} http://192.168.33.10 -o /dev/null
21         if [ $? -eq 0 ]; then
22           echo '. curl Success'
23         else
24           echo '. curl Fail'
25         fi
26       register: result
27
28 4- name: debug
29   debug:
30     msg: "Curl status: {{result.stdout}}"
31
32
```

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

./roles/nginx/handlers/main.yml



```
1 1|- name: restart nginx
2   service:
3     name: nginx
4     state: restarted
5   become: yes
6
```

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
1 tomcat_version: '8.5.12'
2 tomcat_home: '/opt/tomcat'
3 tomcat_user: tomcat_as
4 tomcat_group: tomcat_as_group
5 tomcat_unarchived: 'apache-tomcat-{{tomcat_version}}'
6
7 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`

```
main.yml
1 java_version: "1.8.0"
```

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 plugin: Install Process\nLoading mirror speeds from cached hostfile\n * base: ftp.byfly.by\n * extras: ftp.byfly.by\n * updates: ftp.byfly.by\nResolving Dependencies\n--> Ru\nPackage java-1.8.0-openjdk x86_64 1:1.8.0-121-0.b13.el6.x86_64 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", "rc": 0, "start": "2017-03-26 15:41:43.589097", "rc": 0, "start": "2017-03-26 15:41:42.855424", "stdout_lines": ["200. curl Success"], "warnings": ["Consider using curl running curl"]}\n [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]$ █

```

```

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} http://192.168.33.10:8080/; if [ $? -eq 0 ]; then\n echo '. curl Success'\n else\n echo '. curl Fail'\n fi", "delta": "0:00:00.000000000", "rc": 0, "start": "2017-03-26 15:42:52.771560", "stderr": "", "stdout_lines": ["200. curl Success"], "warnings": ["Consider using get_url or uri module rather than 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=22   changed=8    unreachable=0    failed=0

[student@epbyminw2473 day-2]$

```

```

[student@epbyminw2473 day-2]$ curl 192.168.33.10

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <title>Apache Tomcat/8.5.12</title>
    <link href="favicon.ico" rel="icon" type="image/x-icon" />
    <link href="favicon.ico" rel="shortcut icon" type="image/x-icon" />
    <link href="tomcat.css" rel="stylesheet" type="text/css" />
  </head>

  <body>
    <div id="wrapper">
      <div id="navigation" class="curved container">
        <span id="nav-home"><a href="http://tomcat.apache.org/">Home</a>

```

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.