|  |
| --- |
| EPAM Systems, RD Dep., RD Dep. |
| MTN.\*NIX.11 Automated environment configuration management  **Ansible. 1** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| REVISION HISTORY | | | | | |
| Ver. | Description of Change | Author | Date | Approved | |
| Name | Effective Date |
| <1.0> | Initial revision | Siarhei Beliakou | 16-Mar-2017 |  |  |
|  |  |  |  |  |  |

# Lab Work Task. Tomcat AS Provisioning

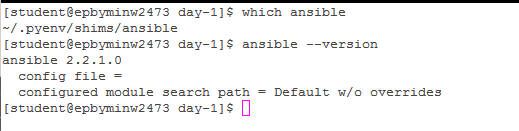
# Review

Using Ansible v2.2.1 for provisioning tomcat application stack. Learning by doing.

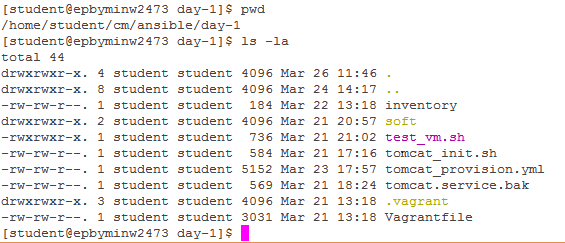
# Task

On Host Node (Control Machine):

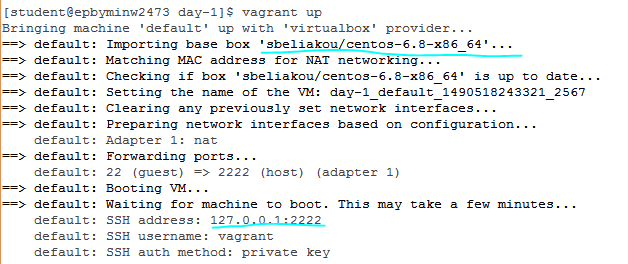
1. Install Ansible v2.2.1 with python pip. Report details where ansible has been installed.



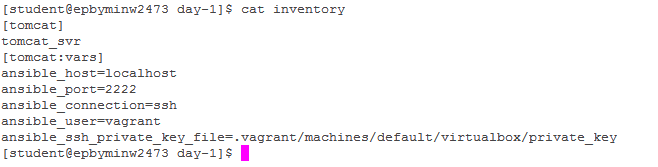
1. Create folder ~/cm/ansible/day-1. All working files are supposed to be placed right there.



1. Spin up clear CentOS6 VM using vagrant (“vagrant init sbeliakou/centos-6.8-x86\_64”). Verify connectivity to the host using ssh keys (user: vagrant)

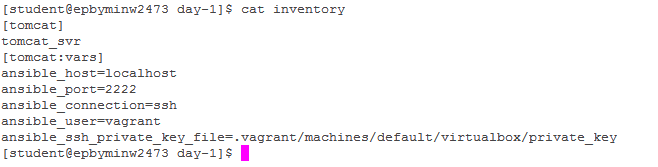


C:\Users\sunof\AppData\Local\Microsoft\Windows\INetCache\Content.Word\3.1.png



1. Create ansible inventory file (name: **inventory**) with remote host connection details:

* Remote VM hostname/ip/port
* Remote ssh log in username
* Connection type

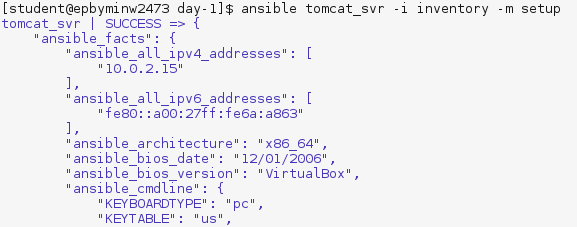


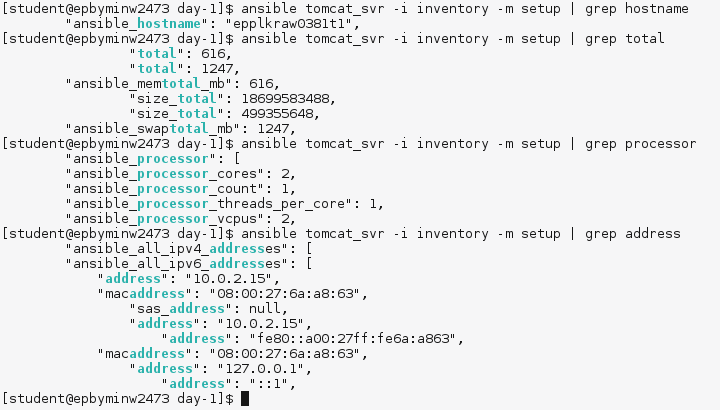
1. Test ansible connectivity to the VM with ad-hoc command:

**$ ansible VM-name -i inventory -m setup**

Find out host details:

* Number of CPUs
* Host name
* Host IP(s)
* Total RAM





1. Develop a playbook (name: **tomcat\_provision.yml**) which is supposed to run against any host (specified in inventory)

Use following modules (at least):

* **copy**
* **file**
* **get\_url**
* **group**
* **service**
* **shell**
* **unarchive**
* **user**
* **yum**

Define play variables (at least):

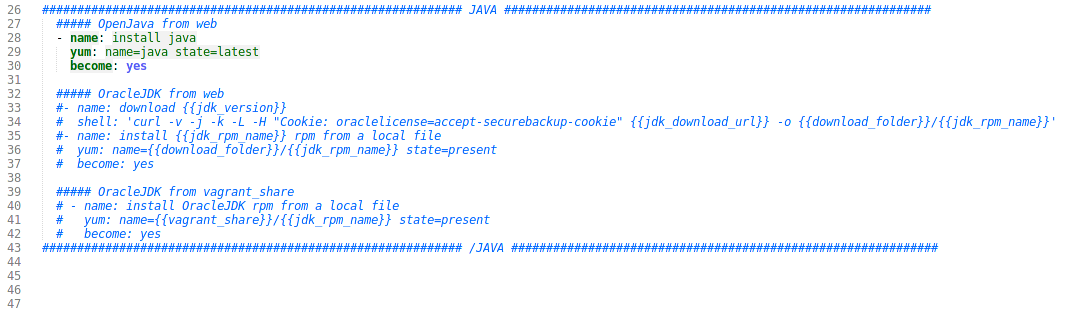
* **tomcat\_version**
* **java\_version**

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

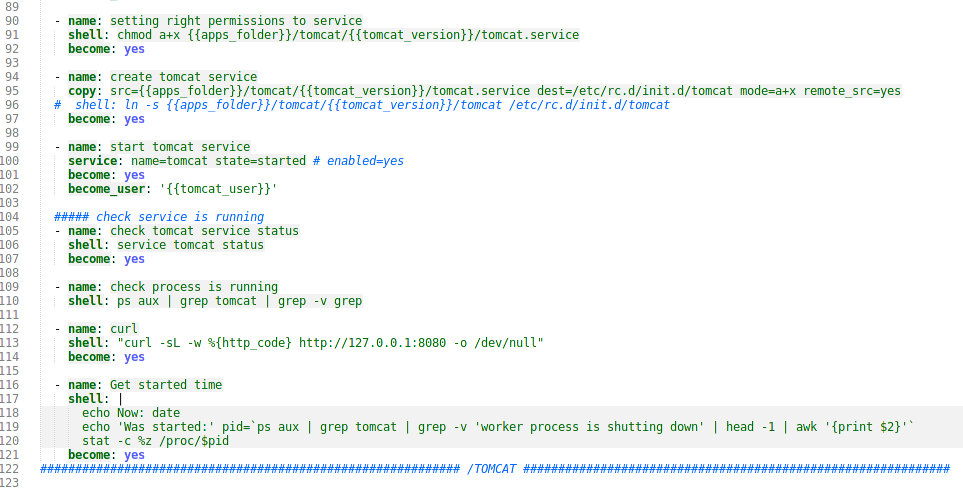
Examples:

* name: Ensure user student exists
* name: Fetch artifact form the Shared repository

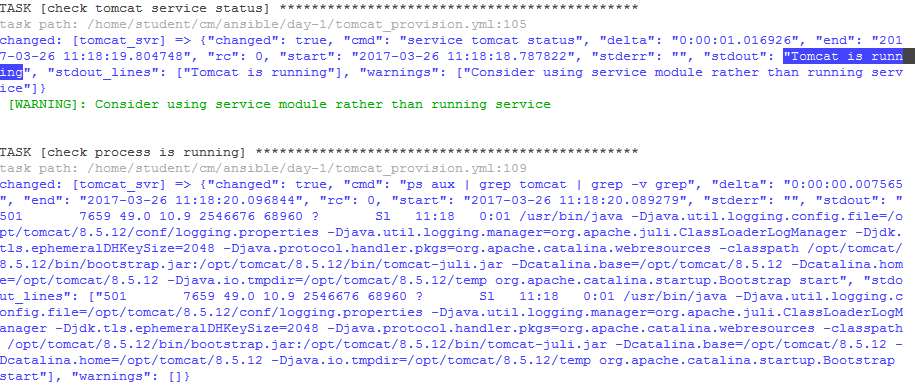


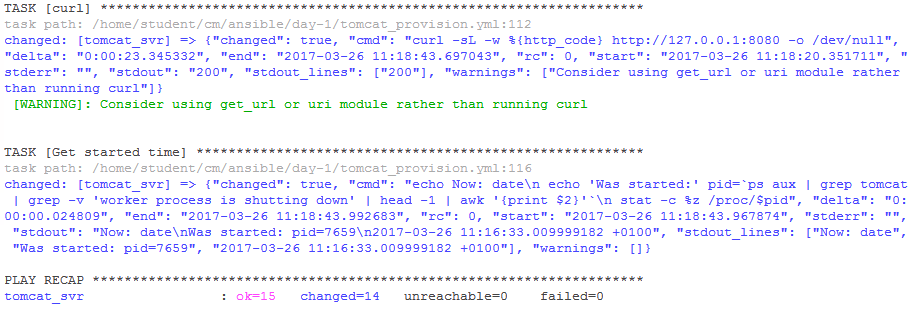




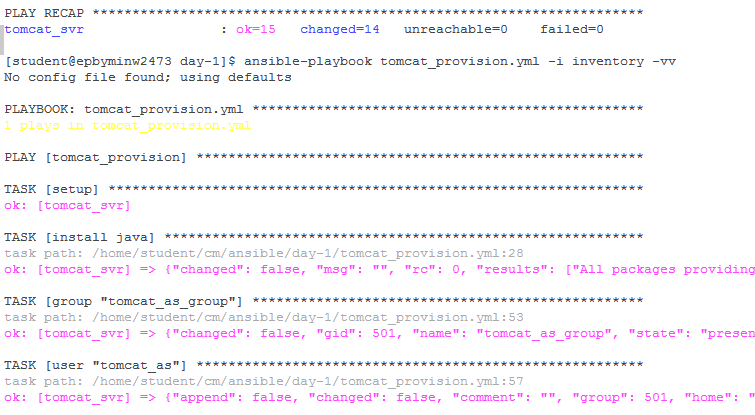


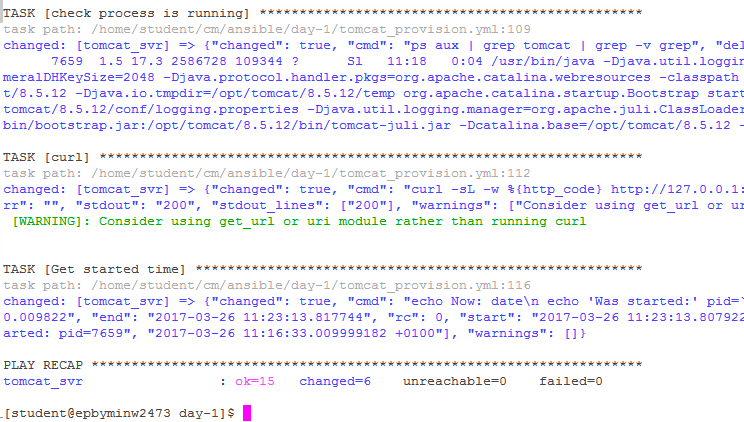
Ensure tomcat is up and running properly with module “shell” (at least 3 different checks).





Second (and other) run(s) of playbook shouldn’t interrupt the service – one of checks should show tomcat uptime.





1. 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 tomcat\_as:tomcat\_as\_group
* Tomcat AS version should be 8.x
* 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

1. Verification Procedure: playbook will be checked by instructor’s CI system as follows:
   1. Connect to student’s host by ssh (username “student”) with own ssh key.
   2. Check the version of ansible installed on the system (as mentioned in point 1)
   3. Go into the folder mentioned in point 2
   4. Destroy/Launch VM: vagrant destroy && vagrant up
   5. Execute VM provisioning: ansible-playbook tomcat\_provision.yml -i inventory -vv
   6. If previous steps are done successfully, instructor will check the report
2. Feedback: report issues/problems you had during the development of playbook and time spent for development.