Contents

[Activity 1: Setting up Ansible 1](#_Toc517775183)

[Activity 2: Creating Inventory file 4](#_Toc517775184)

[Activity 3: Customizing inventory file 5](#_Toc517775185)

[Activity 4: Declaring variables 8](#_Toc517775186)

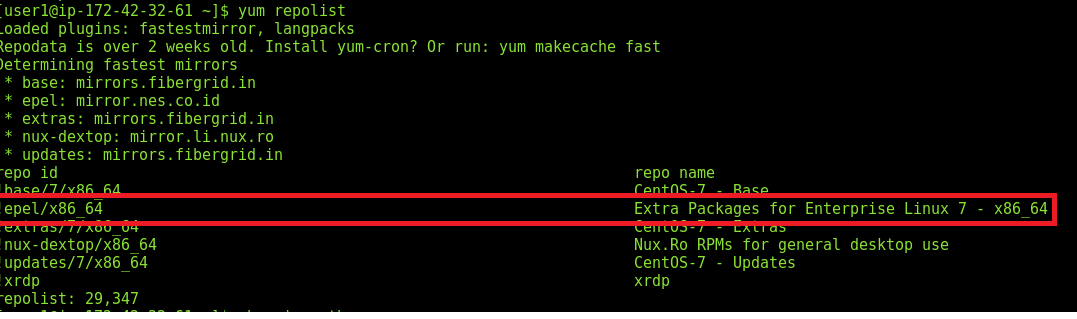
[Activity 5: Installing in the node using modules 10](#_Toc517775187)

## Activity 1: Setting up Ansible

Ansible need to be installed only in the control machine. The nodes need not have ansible installed. The requirement for Ansible to be implemented is

1. **Python present in all machines involved**
2. **Ansible installed in the control machine.**
3. **Able to SSH from control machine to the nodes.**
4. **To make sure Python is installed, do the following:**



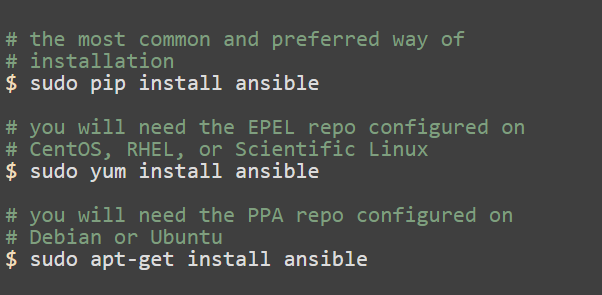


To confirm python is installed do the below:

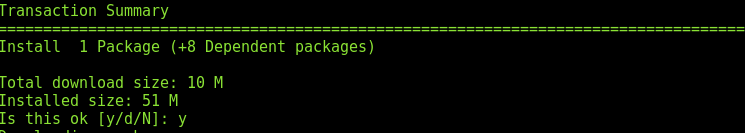




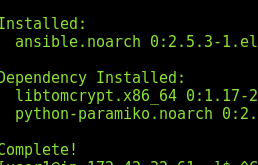
1. **Go to the control machine and install ansible:**



After you run the above command, you will be asked for this confirmation as below:

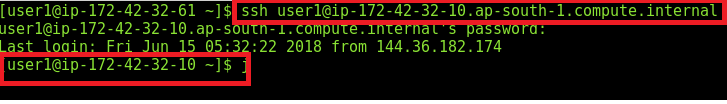


Once it is installed, the following will be displayed:

sudo

1. Configure the Control machine to be able to SSH node:

When you try the SSH you should get the below:

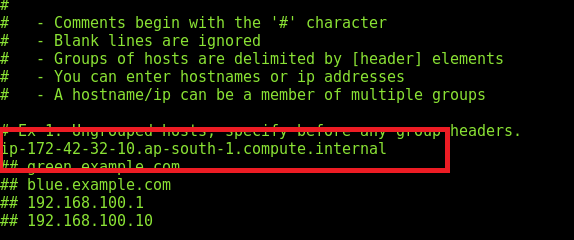


Go to /etc/ansible/hosts which is ansible’s default inventory file:

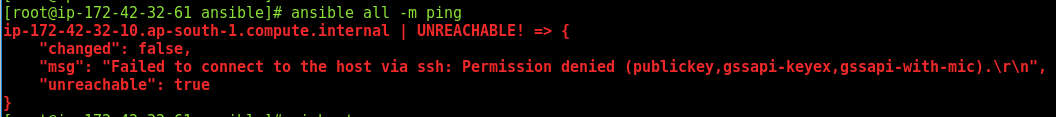
sn

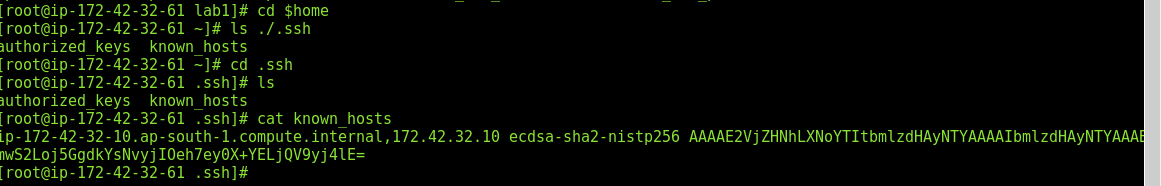


Below is the addition of nodes(hosts) into the control machine’s hosts file.



The below command uses the ping module in Ansible and pings all the hosts(nodes) from the Control node.



* If you get the above error, make the **passwordauthentication = yes** in the host configuration.
* You need to have SSHed before trying the ping module, as the Control Machine needs to have the hosts/nodes fingerprints. The same can be verified with below:
* 

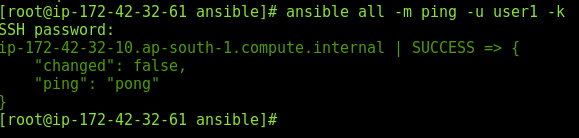
After verifying, use the ping module to ping into all the nodes/hosts mentioned in the /etc/ansible/hosts file.

:q!

Clear

:q!

-k option asks for the password to connect to the hosts.



## Activity 2: Creating Inventory file

1. Create a directory called lab1 inside the ansible directory



1. Navigate inside the directory and create an inventory file.



1. Add the host(s)/node(s) into this inventory file:



Note: ANSIBLE INVENTORY CONFIGURATION: You can create more than one inventory file and call any at the command line. For ex: one for test, one for dev instead of one long inventory.

Ansible all -i inventory -m command -a “/usr/sbin/**yum update -y**”

The above command will help update all your nodes mentioned in the file named “inventory”. Here “inventory” is the filename of your inventory file. -i is the parameter to pass the inventory file.

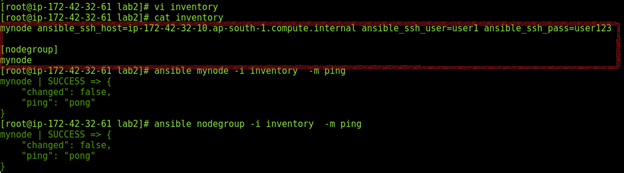
## Activity 3: Customizing inventory file





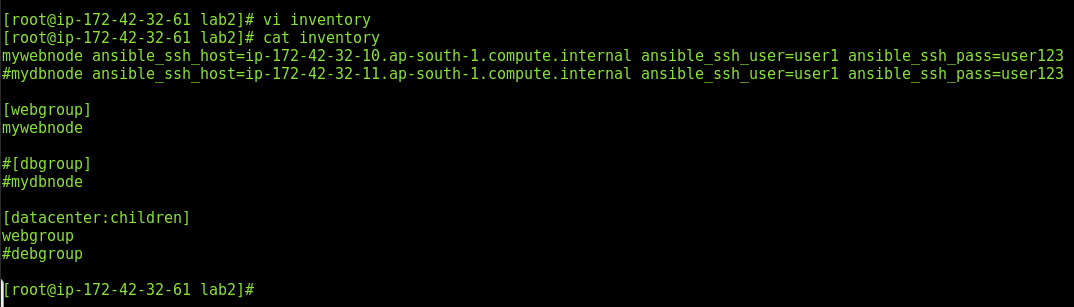
1. Name the node as “mynode” and define it under the group [nodegroup] as highlighted.

Post which in your command you can use a particular node name instead of “all” against which the ping module needs to be run:

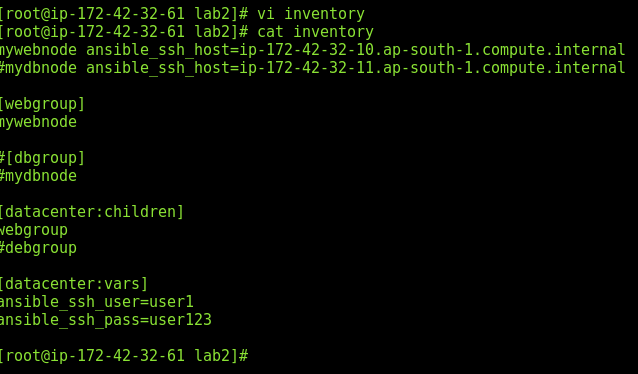


Note: One node can belong to multiple groups.

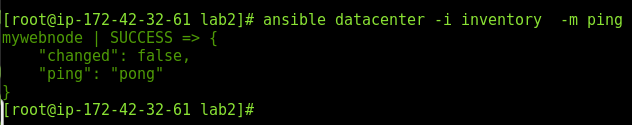
1. You can edit the inventory file further more, to group various nodes under various groups and having a group of groups. In the example below it is called “datacenter”. Not the keyword “children” mentioned.



1. To avoid the password being passed for every node, if it is the same password shared across nodes then add it in variables as below, which will be applied to all the nodes inside of “datacenter”



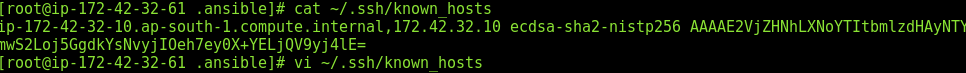
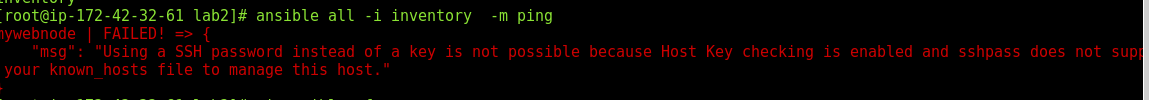
Execute the ping module across all nodes under the datacenter as below:

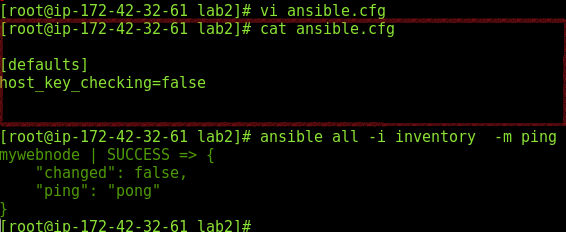


1. Introducing ansible.cfg file.

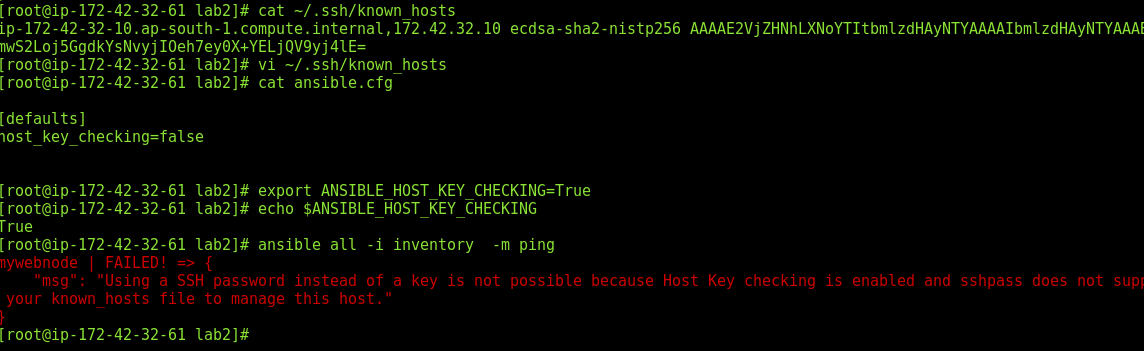
Certain settings in Ansible are adjustable via a configuration file (ansible.cfg).

For example:

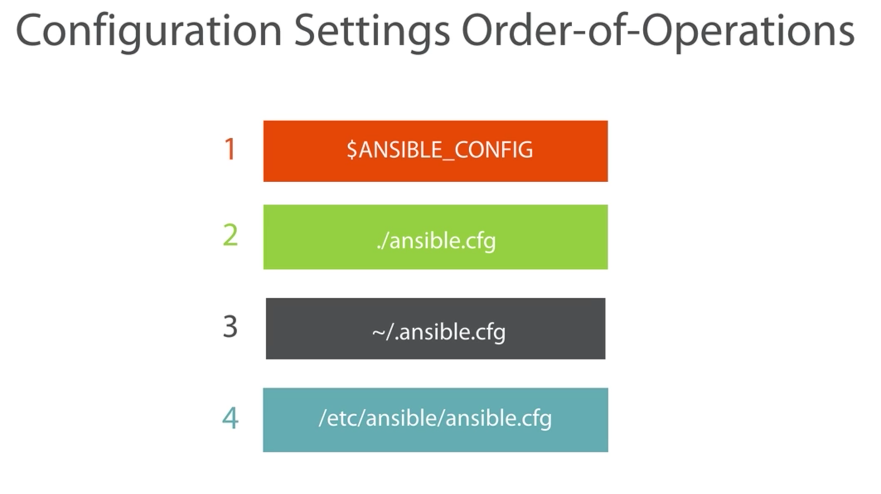
* Create an ansible.cfg file inside lab2 folder using command “touch ansible.cfg”
* Go to known\_hosts
* And delete the content above related to your host/node
* Try using the ping module and observe the below error
* To resolve it add the ansible.cfg, in the lab2 folder with the below contents, to disable the verification of known hosts.



* Verify now again in known\_hosts, will be auto populated, as it has connected the host once.
* *Delete the content of known\_hosts – once again.*
* Now set the environment variable of “ANSIBLE\_HOST\_KEY\_CHECKING” to True as below



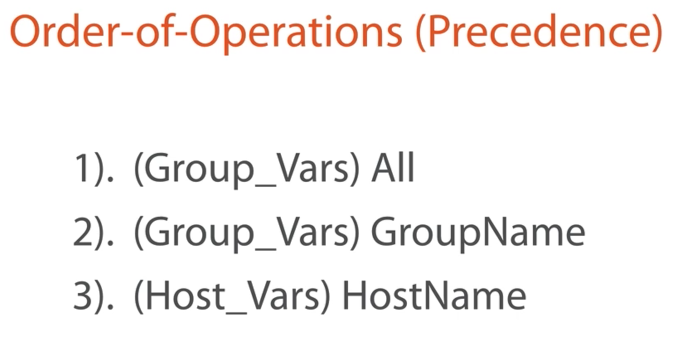
* Now execute the ping module, and observe that it fails inspite of what is set in the ansible.cfg file.
* This shows how environment variable takes precedence over the local variable.
* The precedence is as below: The 3 files are mutually exclusives; if one exists (in order of precedence ) the others are ignored completely, even if the sections variables are non overlapping.
* Changes can be made and used in a configuration file which will be searched for in the following order:



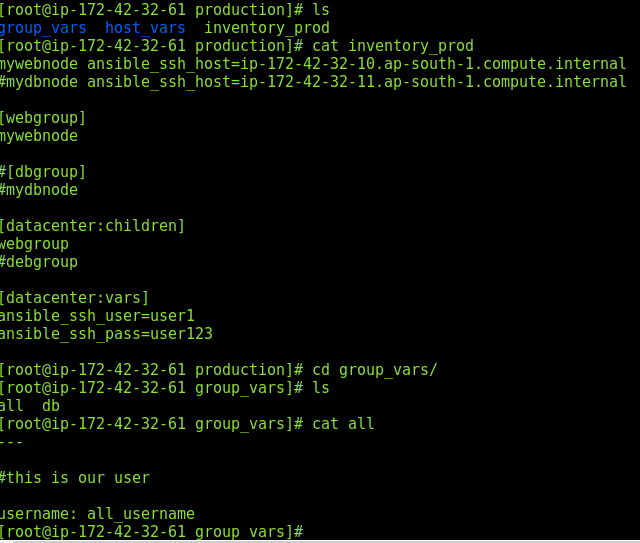
Changes can be made and used in a configuration file which will be searched for in the above order. Ansible will process the above list and use the first file found, all others are ignored.

## Activity 4: Declaring variables

* To understand GROUP\_VARS, create directory structure as mentioned below under LAB 3. The blue highlighted ones are folders and the green ones are files.
* Though we would not be using all the files & folders, this activity is to just get used to creating directory structure for defining group variables.
* The order of precedence is as below: ( 3 being the highest ) ie from low to high



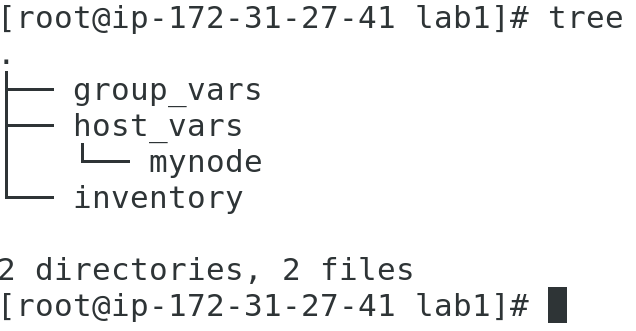
1. Create directory structure as shown:
2. Use tree command to view the same. We would be using only production.
3. If tree is not present, then use “yum install tree” to install
4. Cd production and create the inventory\_prod inventory file as in the previous exercise.
5. Define the username variable value as “all\_username” in the all file In group\_vars.



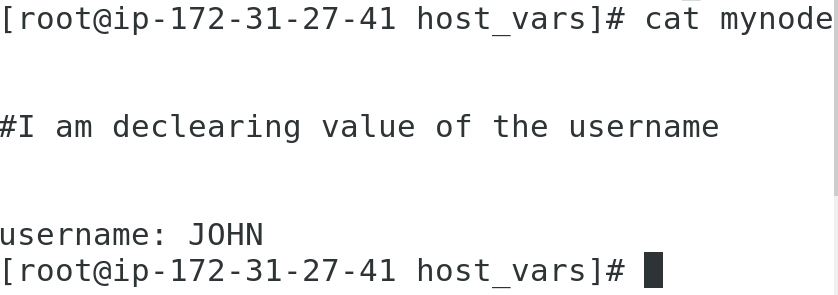
1. Use the below command to create username as mentioned in the variable value in the nodes.



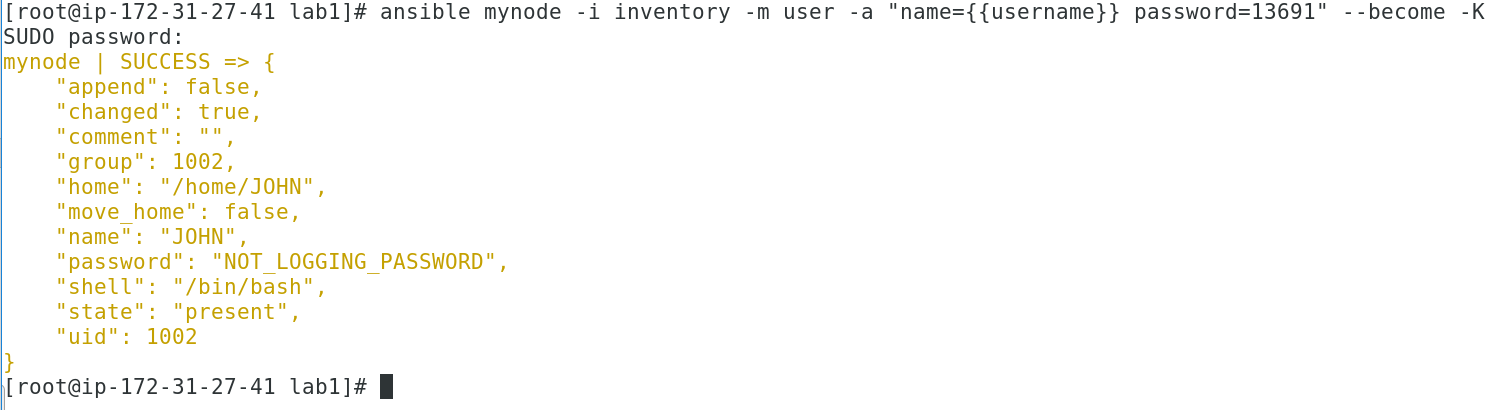
Alternatively: you can simply use the previous lab directory and create the folders like below:



In the file mynode, the variable values need to declared. I have used the variable name as “username”, however it can be any name.



The command to create username in the node is as below:



Now you can connect to your node through remote desktop, go to /etc folder and view the file “passwd”

Cd /etc

Cat passwd

Look for the user that you created.

Likewise, you can try other modules,

To list all the modules:

ansible-doc -l

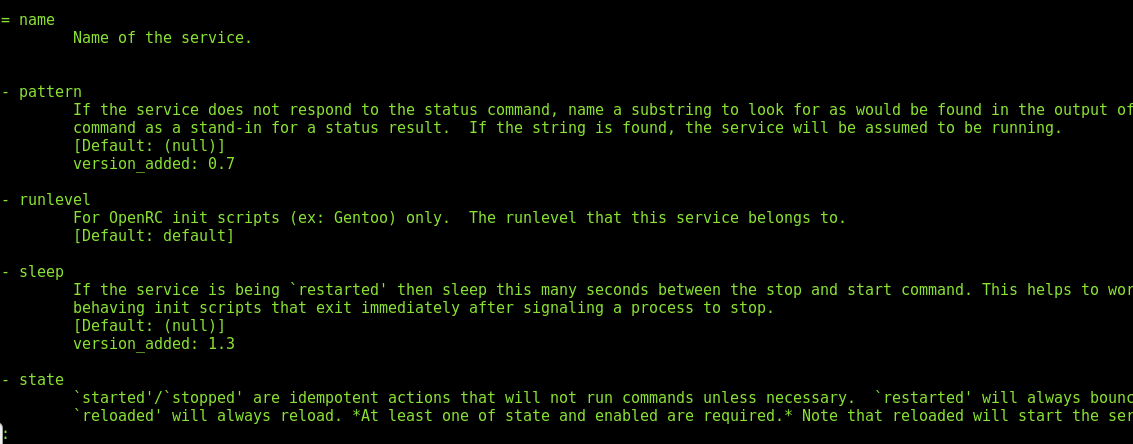
ansible-doc -s <modulename>

ansible-doc <name>

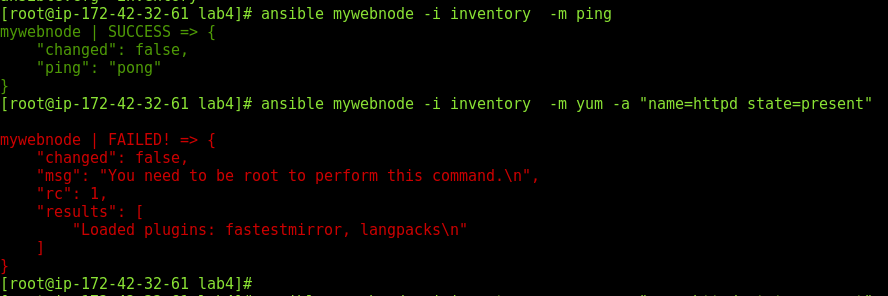
## Activity 5: Installing in the node using modules

1. Use the same inventory file as in the previous activity. Create another folder called lab4
2. Using the yum module and installing apache tomcat service in the node:
3. For the parameters for service refer the document from the below command

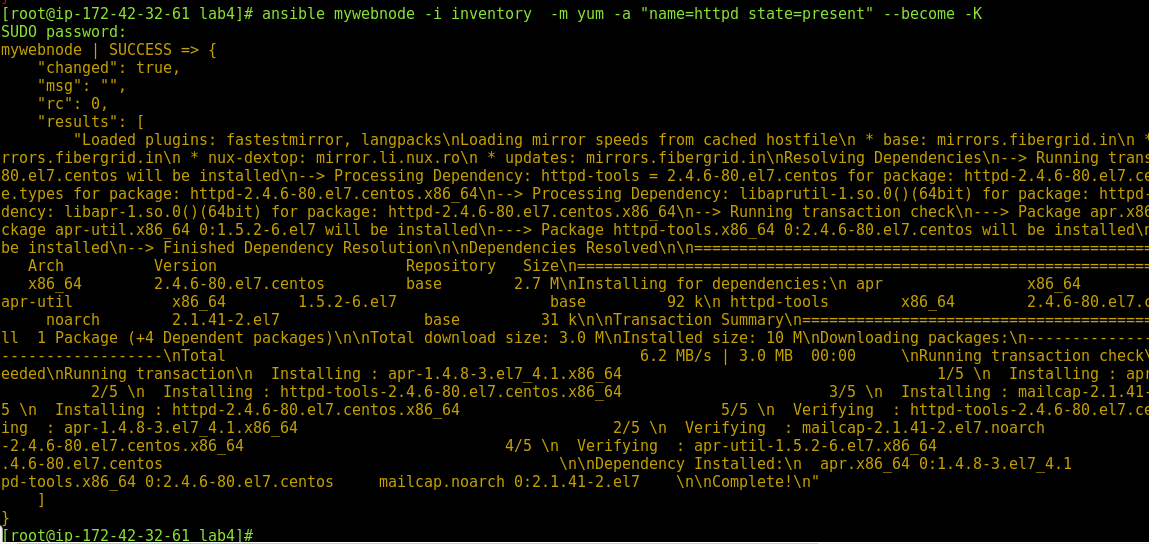
ansible-doc service:



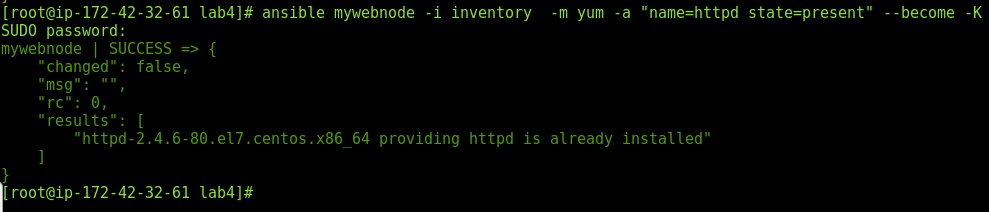
1. Installing the service using yum module:



1. Since installation needs root user access, use “become” to sudo into root

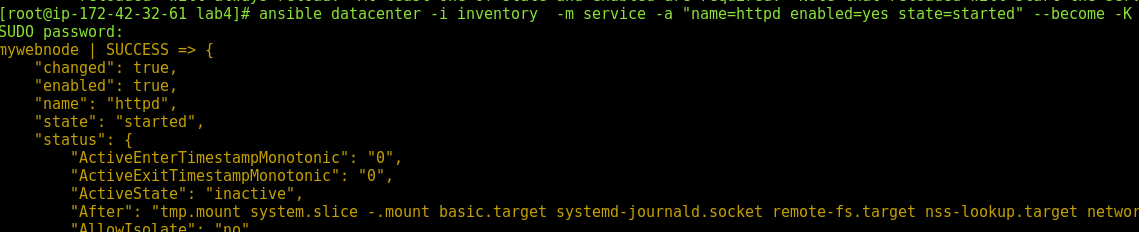


1. If you try to install it again, it would say already installed.

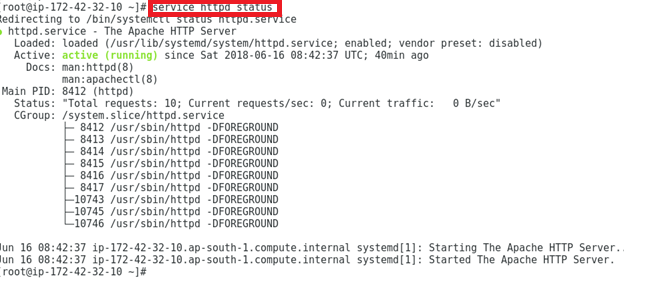


**Note the “changed” value in both, when it installed for the first time “changed” is True, whereas the second time, it didn’t change anything in the node, hence “changed” is false.**

1. Start the apache tomcat service which has been installed in the node



1. Login to the node machine and check whether apache tomcat is installed and running:



1. Also access the IP from the browser in the node and verify apache http server is responding:

