TLEN 5410 – Network Management and Automation

Lab 7

DevOps - Ansible

University of Colorado Boulder

Interdisciplinary Telecom Program

Professor Levi Perigo, Ph.D.

Summary

Ansible has proved to be profoundly different from other configuration management tools available today, and is prevalent in industry for managing and configuring network devices as well as servers. It has been designed to make configurations easy in almost every way. Some of its notable features include its simple English configuration syntax, agentless running service, and its ease of setup.

# Objectives

* Configuring and installing Ansible.
* Learn to use Playbooks for configuration management.
* Learn how to perform package management on the hosts.
* Learn how to perform identity and access management.

* Learn to perform file transfers and directory management.

Problem Statement

Liverpool and Real Madrid have a strong history of European Football. The two teams have once again made their way to an epic Champions League final. The FIFA administration has been entrusted the job to ensure that all things are in place to give us a thriller.

You will be acting as the FIFA administrator and be performing the tasks given below.

# Objective 1 – Setting up a Virtual Network

# 

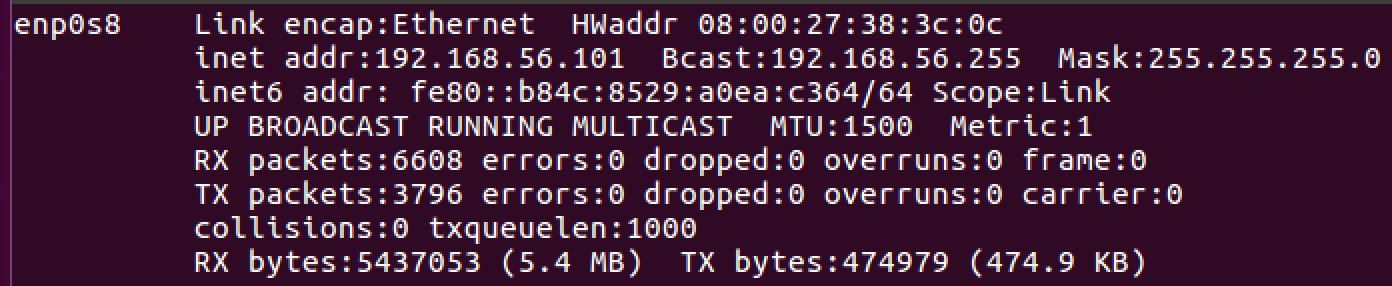
**Figure 1**

1. Create the above network of three virtual machines (use the NetMan VM as the FIFA\_admin, which will act as the Ansible Server) in Virtualbox. You can use any Linux based OS of your choice to spawn the Liverpool and Real Madrid virtual machines.

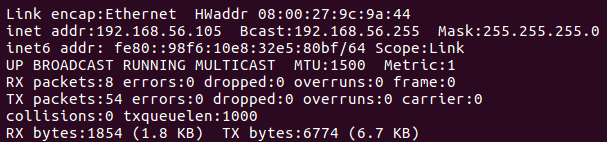
Note: Set up other VMs(Liverpool & Real Madrid) with host-only network adapter

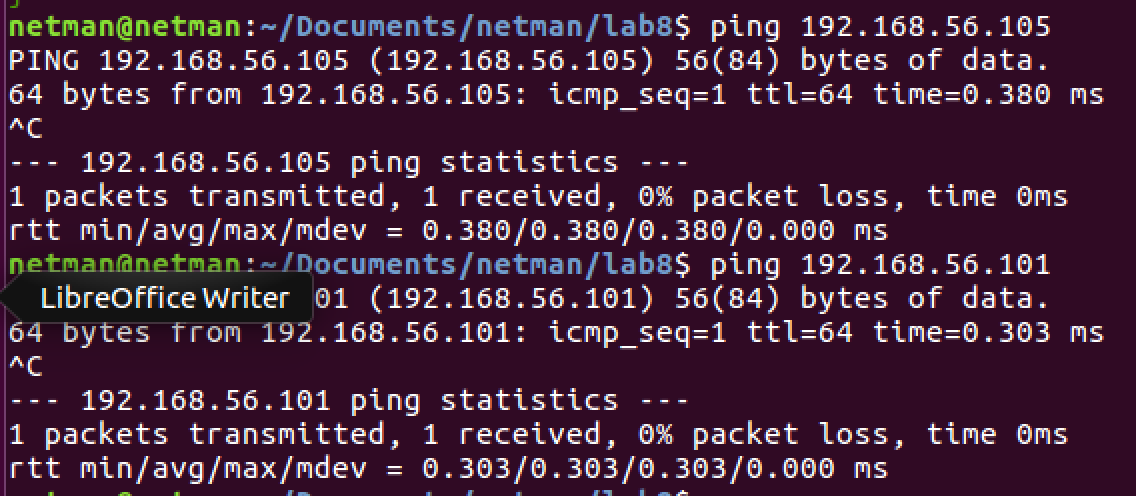
1. Configure IP addresses (of your choice) and establish connectivity between the machines (HINT: configure network adapters as required). Provide screenshots of successful network connectivity. **[10 Points]**

Liverpool:



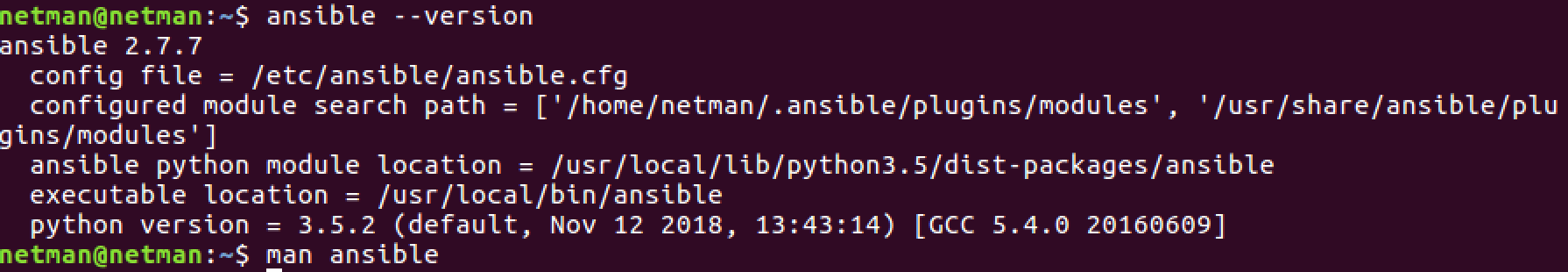
RealMadrid:





# Objective 2 – Installing Ansible

1. Follow instructions from <http://docs.ansible.com/ansible/intro_installation.html>, install Ansible for the control machine (FIFA\_admin). We recommend using Python 2.7 with Ansible.
2. Issue the “**ansible --version”** command to check successful installation. Provide screenshot of the output. **[5 Points]**

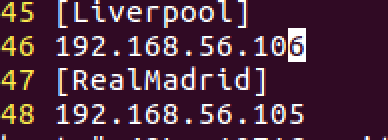


1. In order to allow FIFA\_admin to control the two hosts through a playbook, ensure that you add the hosts in the /etc/ansible/hosts or any other Inventory file of your choice (<http://docs.ansible.com/ansible/intro_inventory.html>). Ensure connectivity between the server and managed hosts using the “**ansible all -m ping --ask-pass**” command. Provide screenshot of the output. **[5 Points]**

In FIFA\_admin:

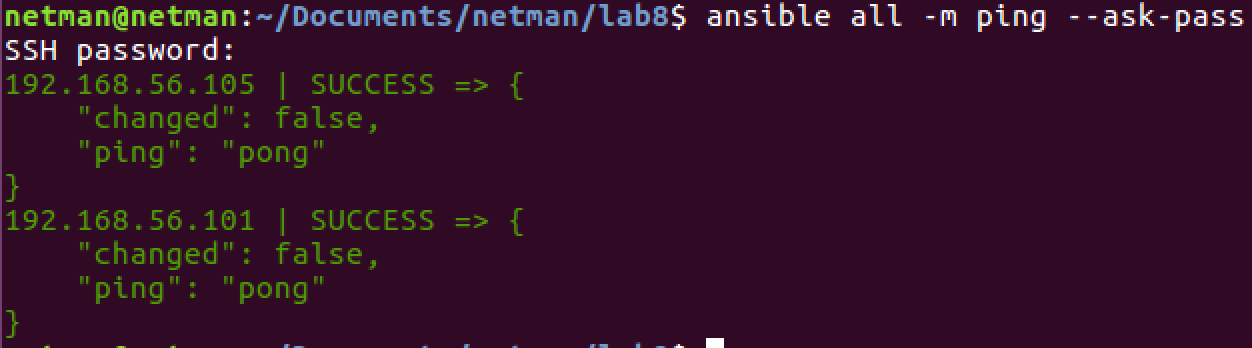


At the button and add the following to add hosts:

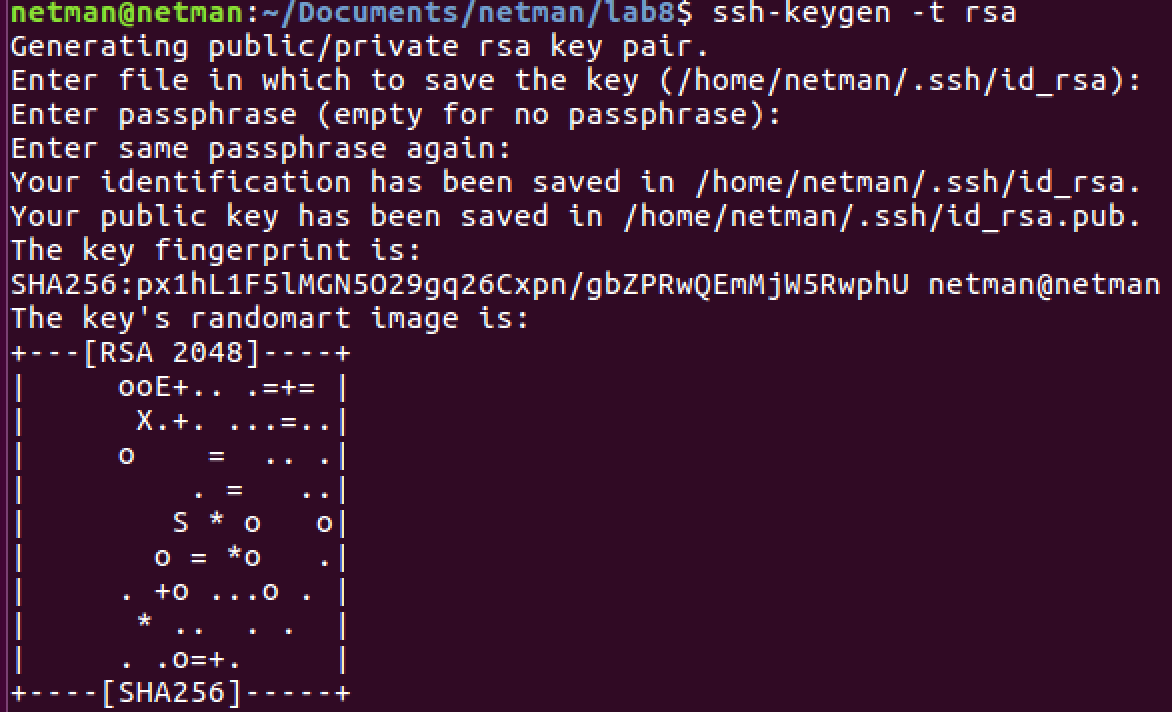


This is without setting SSH from Server to client, manually type ssh password

SSH password: netman



This is with setting ssh from Server to client:







Do both on Liverpool and RealMadrid machines

Objective 3 – Creating a Centralized Playbook

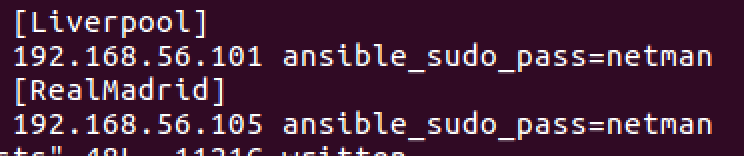
The following objectives need to be accomplished by creating a **single** playbook on the FIFA\_admin machine. The playbook file should be named as “YourIdentiKey\_play.yml”. Make sure you give meaningful names to the tasks performing the below objectives in the playbook.

Objective 3.1 – Installing Packages

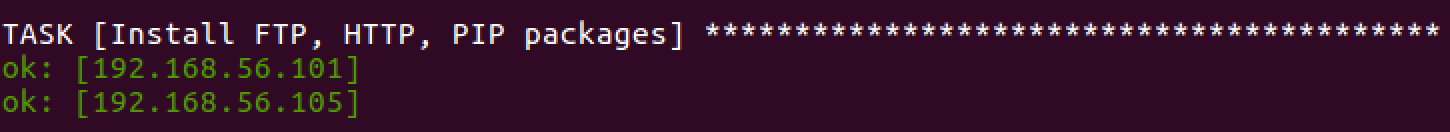
1. Liverpool and Real Madrid machines should have FTP and HTTP capabilities. Additionally they need pip and expect packages. Install all these essential packages through the playbook on FIFA\_admin (use [loops](http://docs.ansible.com/ansible/playbooks_loops.htm) to perform the repetitive installation tasks).

**[10 Points]**

Because I cloned from the class VM, so if I run something that needs root permission, I cannot, so I have to configure password under /etc/ansible/hosts like following:

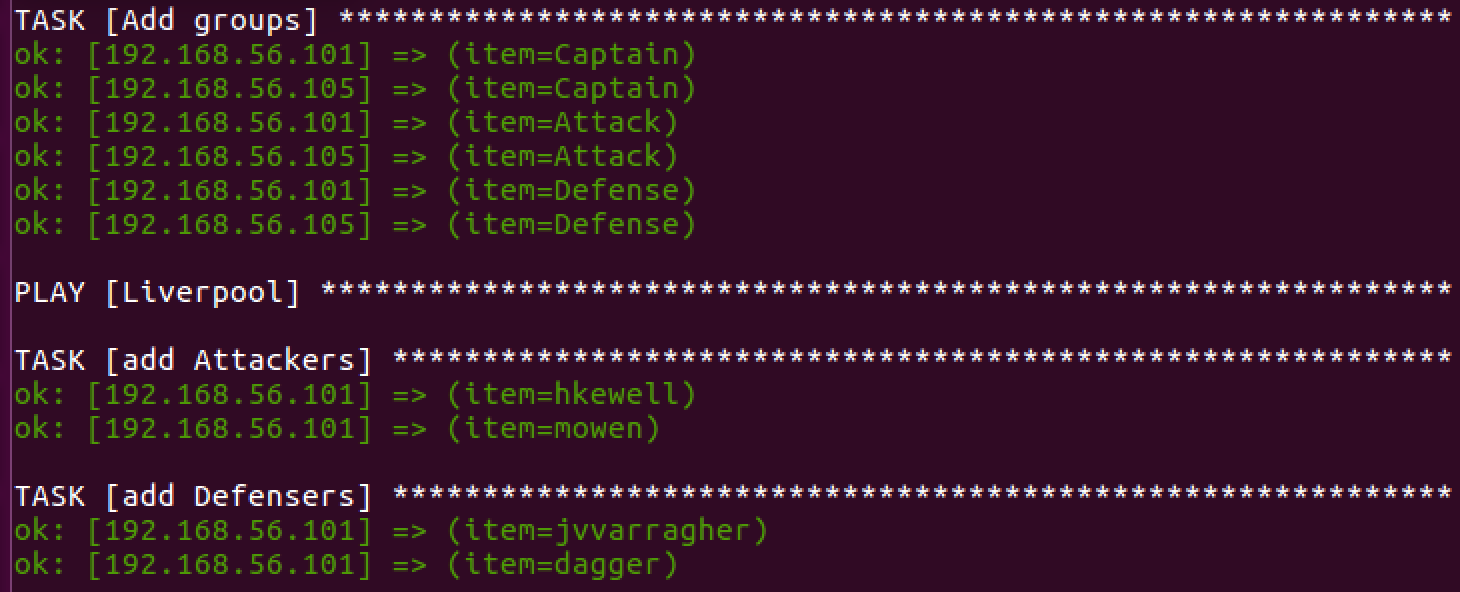


1. Provide screenshots of the relevant portion of the playbook and the successful package installation on the two nodes. **[10 Points]**



Objective 3.2 – Creating Users, Groups, and Permissions

1. Create users and groups on the managed hosts using the same playbook on the control machine. **[5 Points]**



1. The three groups on each machine are: Attack, Defense, and Captain. The users to be created, their corresponding groups, and permissions are mentioned in tables below. The users should be created with the naming format of FirstInitialLastName (for example Harry Kewell’s user name would be hkewell). **[15 Points]**

**Liverpool**

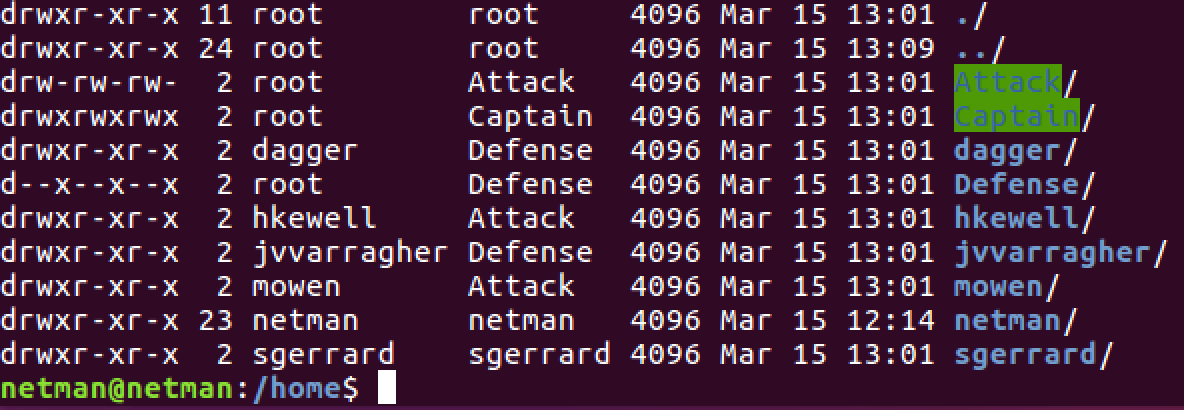
|  |  |  |
| --- | --- | --- |
| **User** | **Group/Groups** | **Group Permissions** |
| Harry Kewell | Attack | rw-rw-rw |
| Michael Owen |
| Jamie Carragher | Defense | --x--x--x |
| Daniel Agger |
| Steven Gerrard (present in two groups) | Captain and Any group with highest access permissions (Eg: Wheel) | rwxrwxrwx |

**Real Madrid**

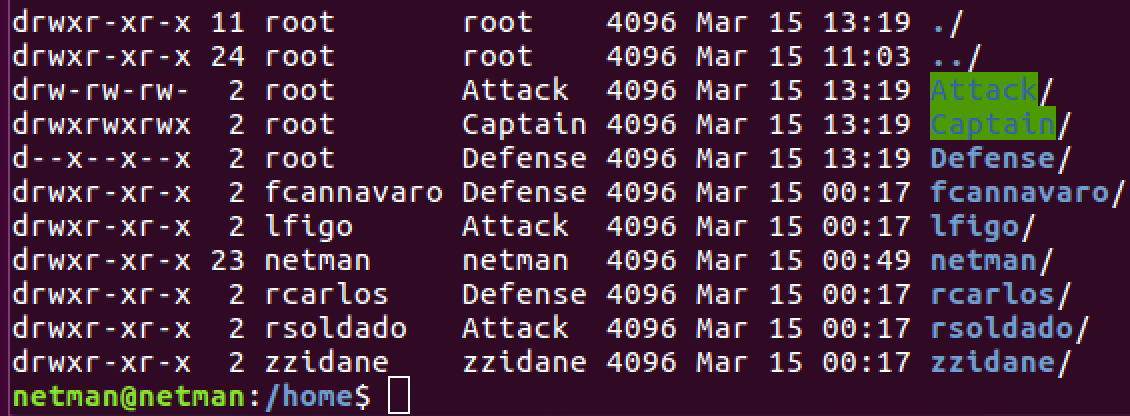
|  |  |  |
| --- | --- | --- |
| **User** | **Group/Groups** | **Group Permissions** |
| Luis Figo | Attack | rw-rw-rw |
| Roberto Soldado |
| Roberto Carlos | Defense | --x--x--x |
| Fabio Cannavaro |
| Zinedine Zidane  (present in two groups) | Captain and Any group with highest access permissions (Eg: Wheel) | rwxrwxrwx |

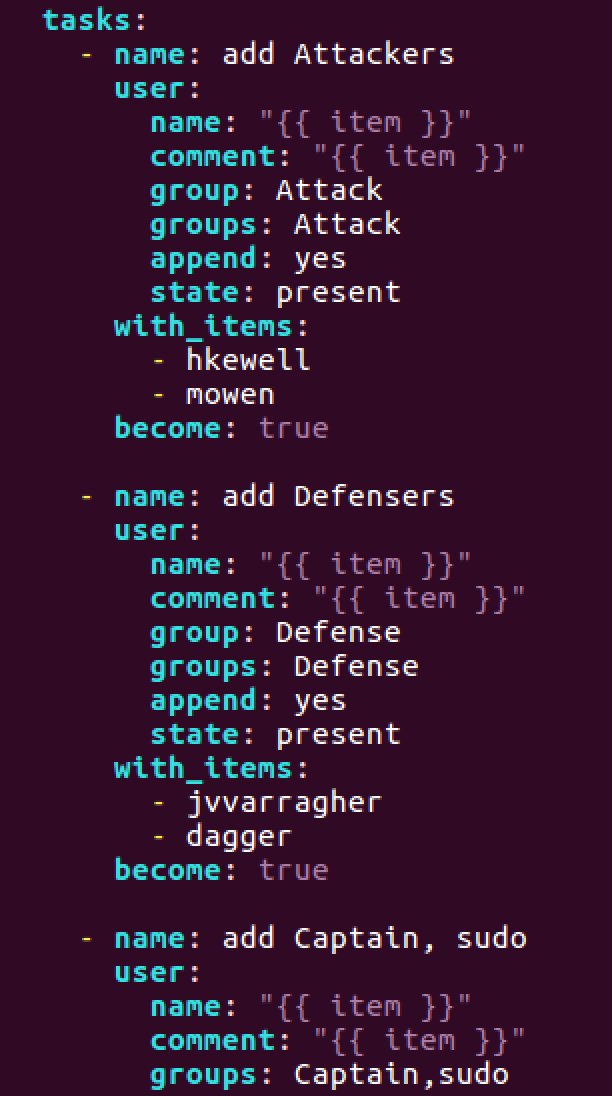
1. Provide screenshots of the relevant portion of the playbook, successful creation of all users and their group permissions. **[5 Points]**

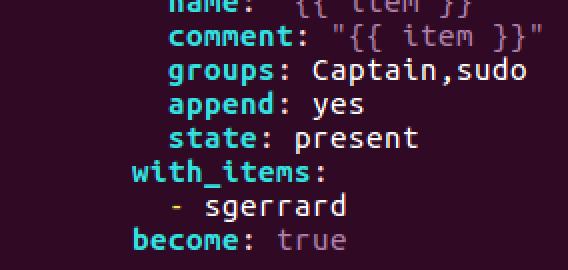
Liverpool:

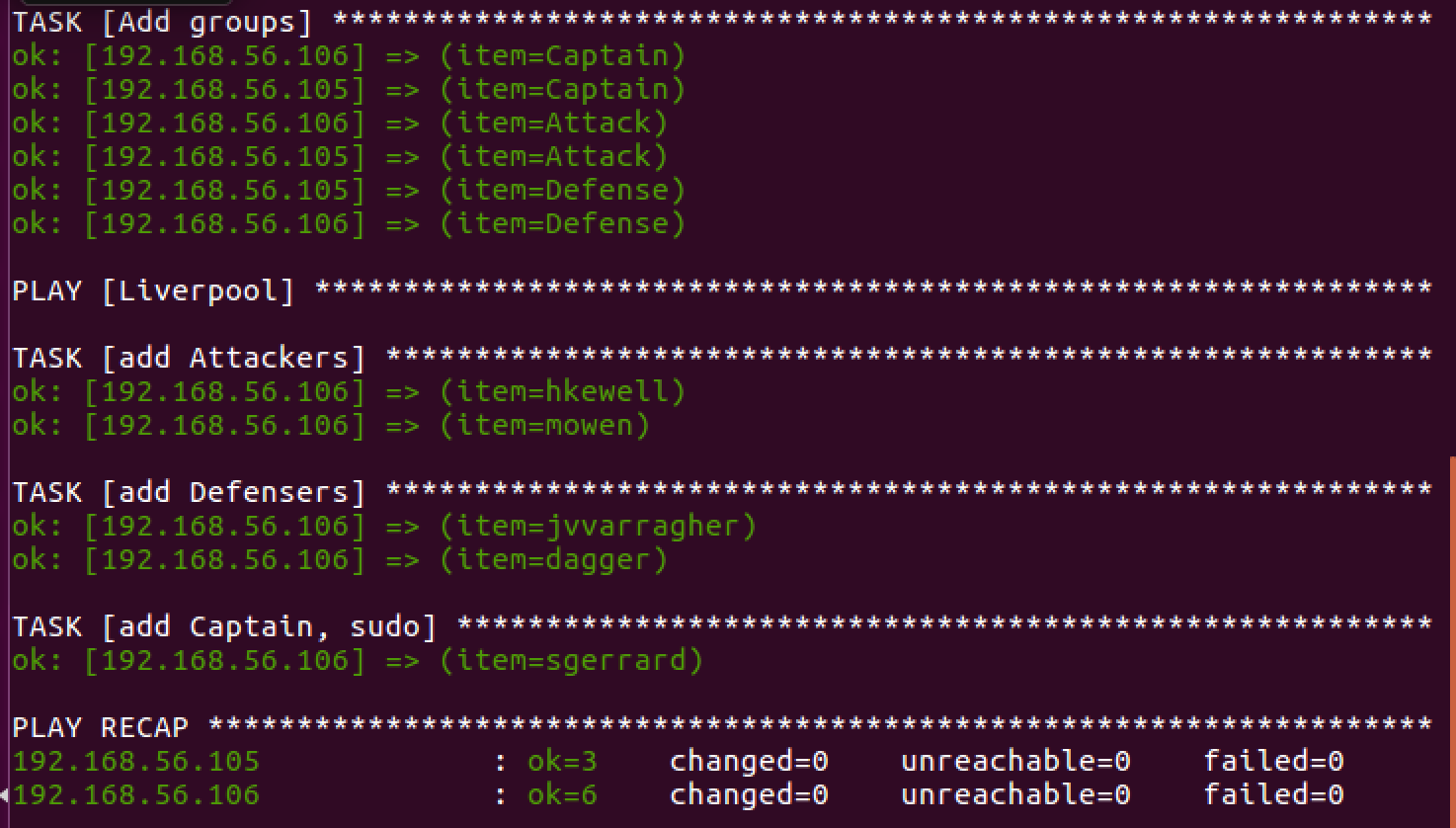


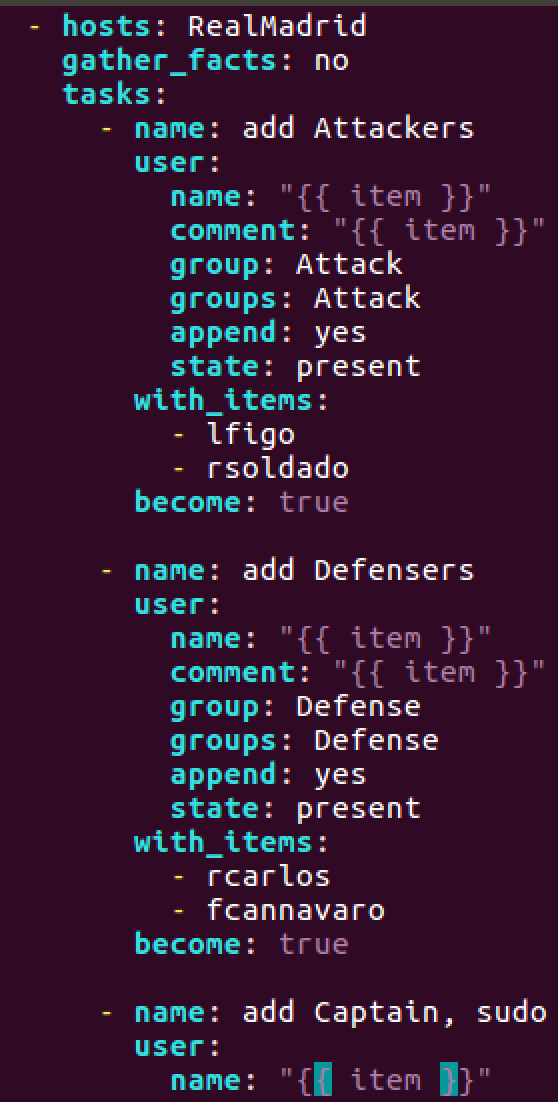
RealMadroid:

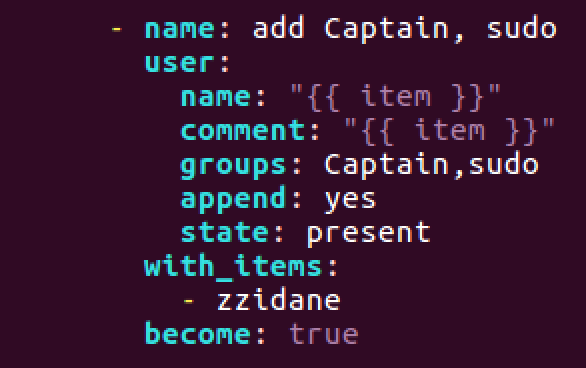


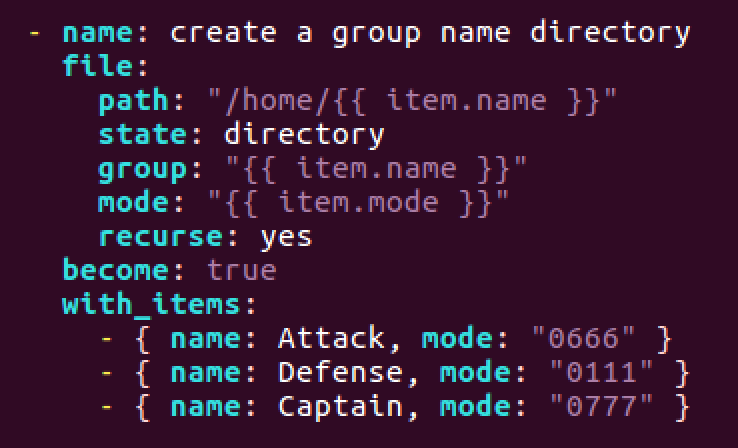


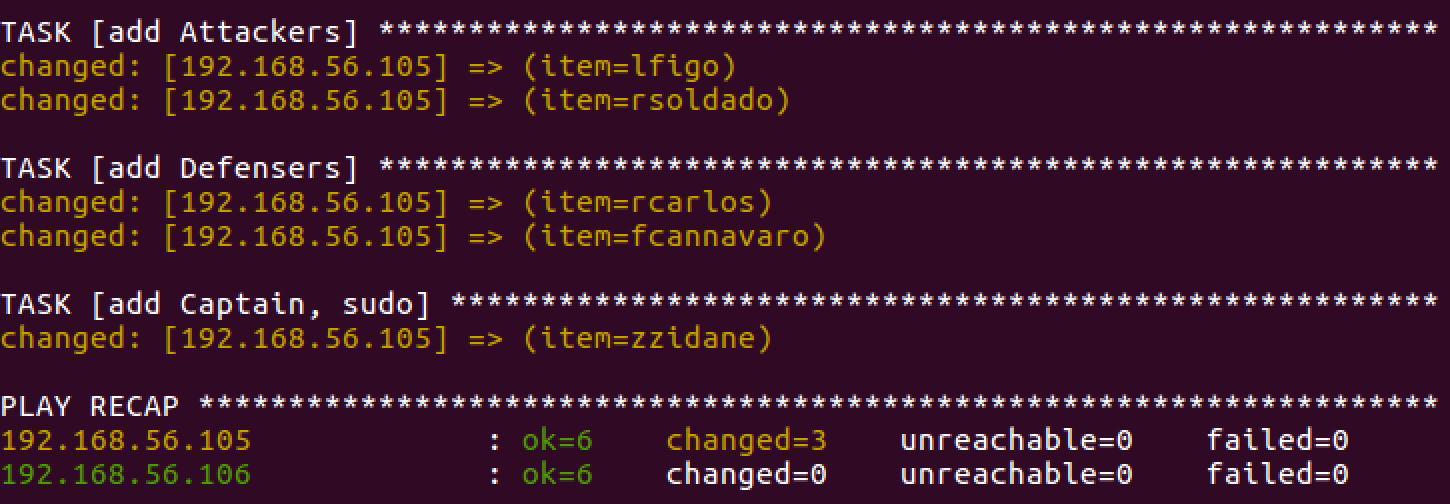


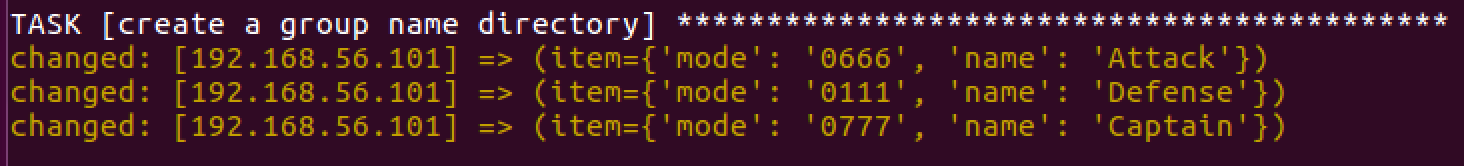


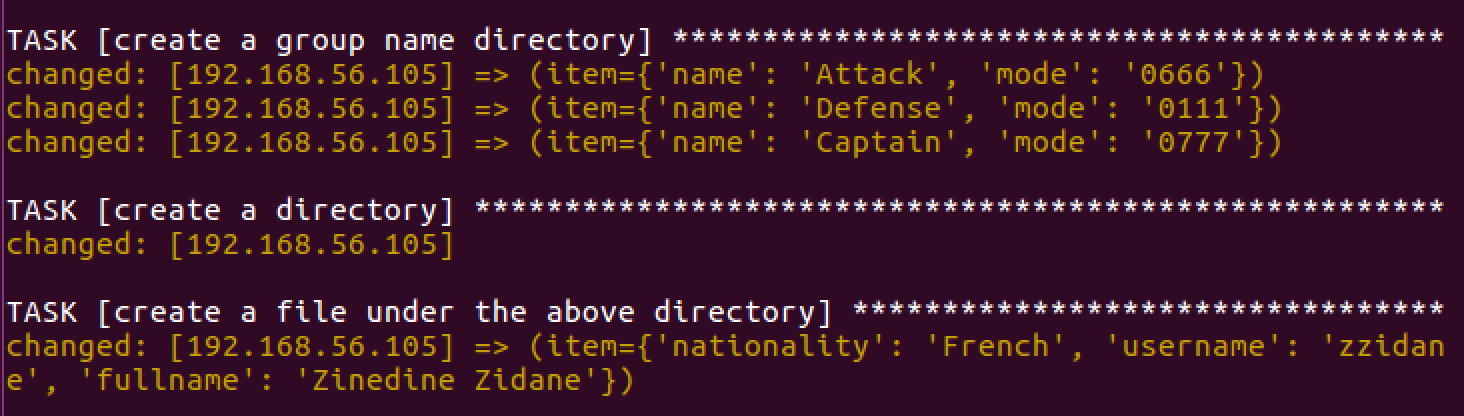






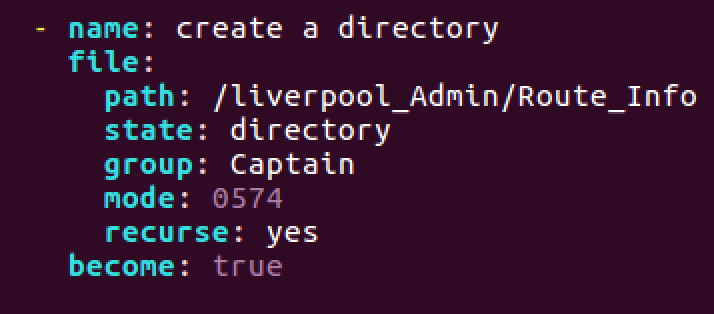
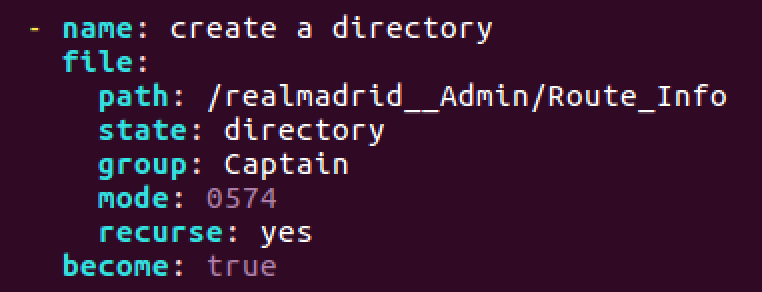


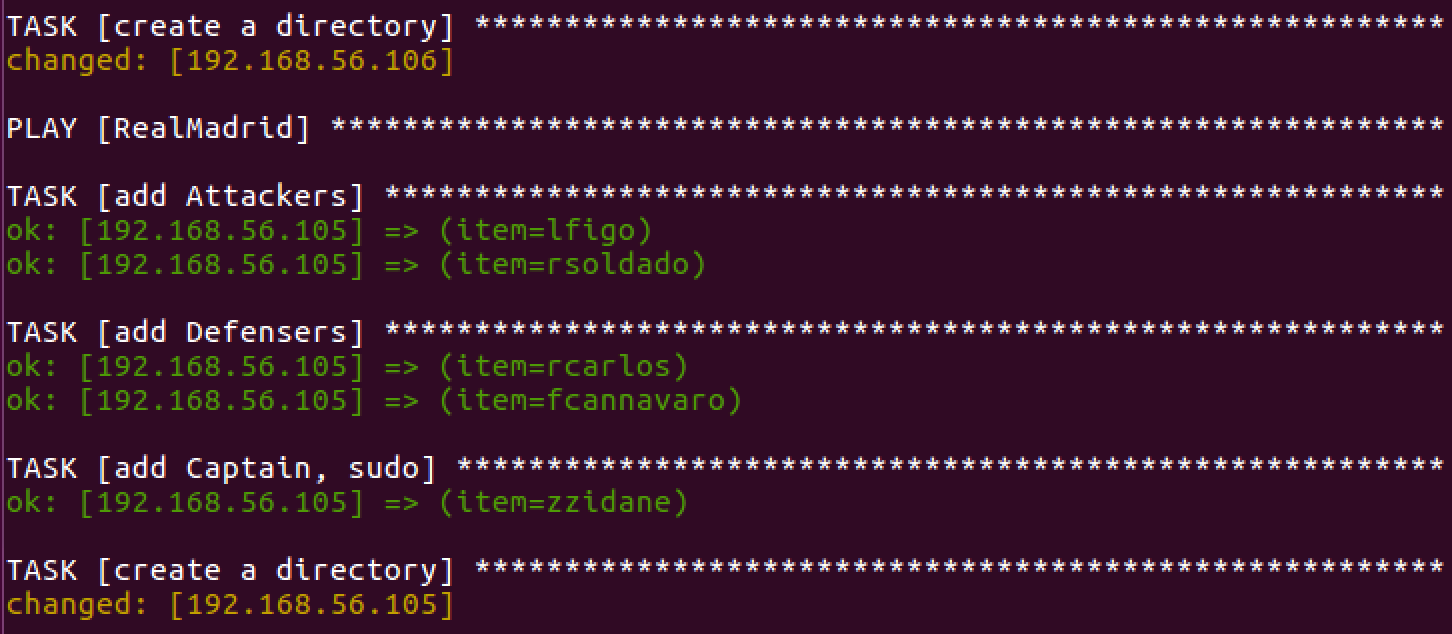




Objective 3.3 – Creating Files and Directories

1. Add commands to the same playbook to create directories on Liverpool and Real Madrid which will be used to store route table information in the next objective. The directory should be created in the path /MachineName\_Admin/Route\_Info/. The owner and group owner for this directory should be Captain, and they should have all permissions. Users who are not in the Captain group should have read-only access to this directory.
2. Provide screenshots of the relevant portion of the playbook, successful creation of the directory and applied permissions. [**10 Points**]

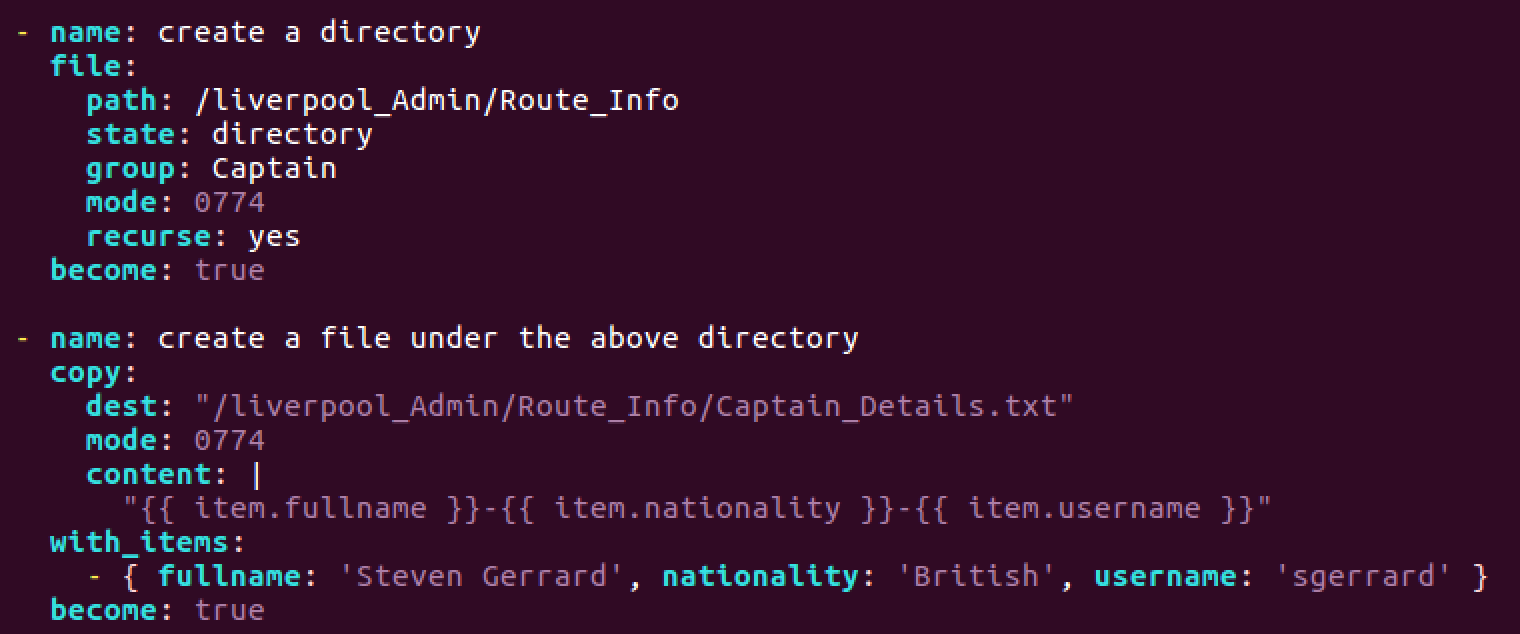


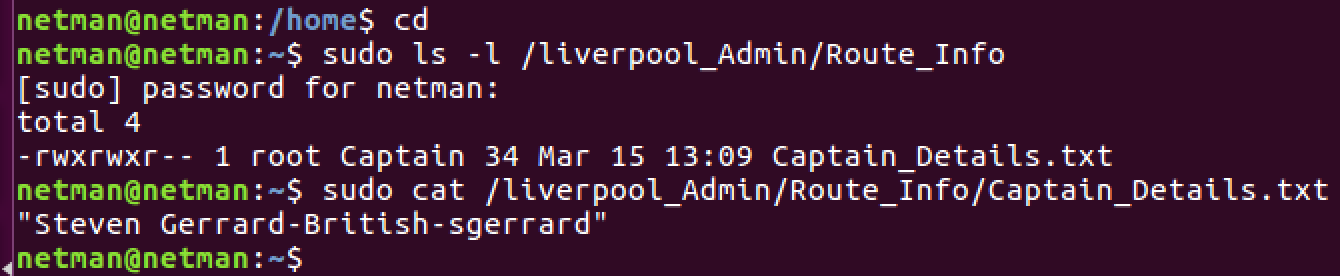


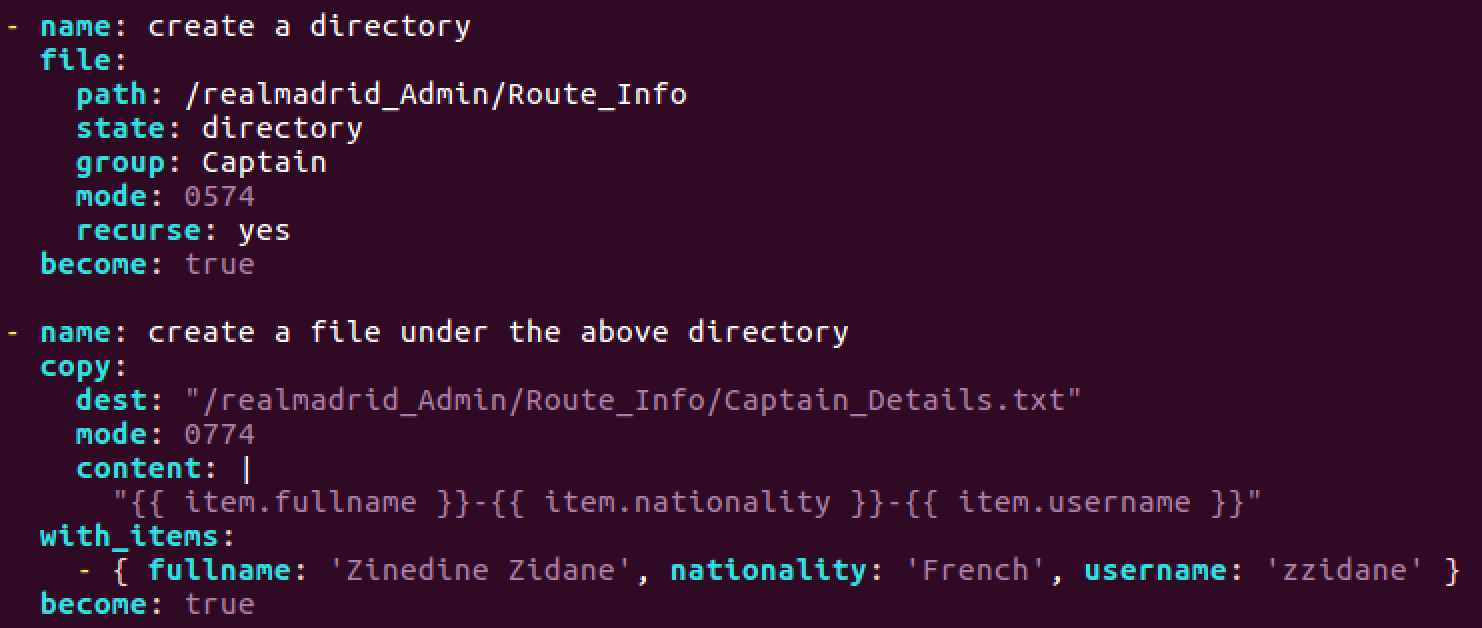


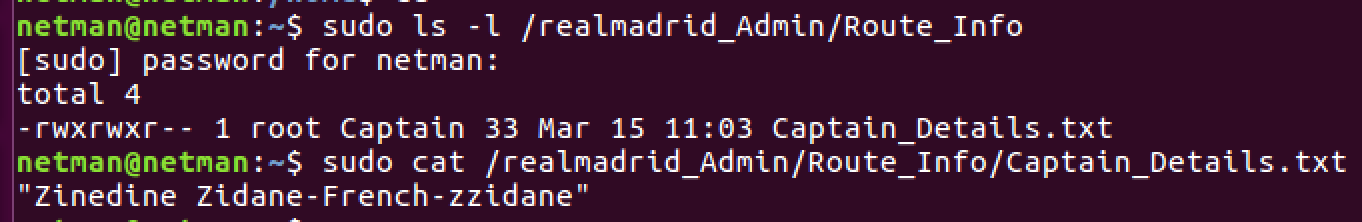


1. On each host, create a file Captain\_Details.txt under the above created directory and store the captain information (full name, nationality, username) using the same playbook. Provide relevant screenshots. [**10 Points**]



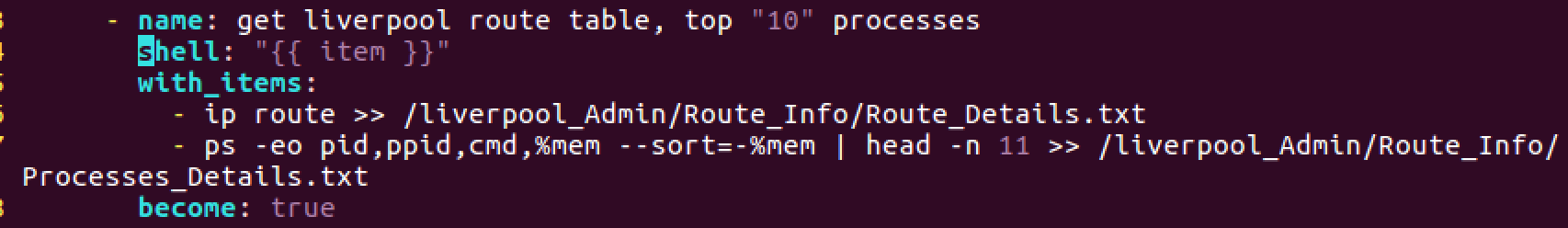


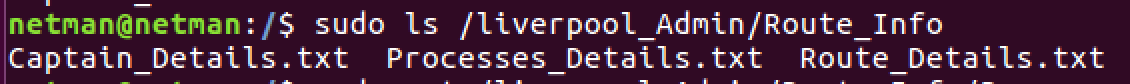


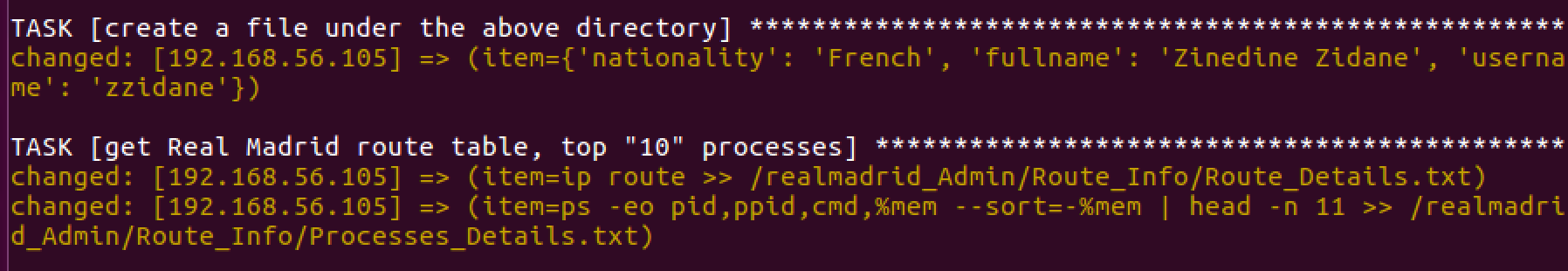


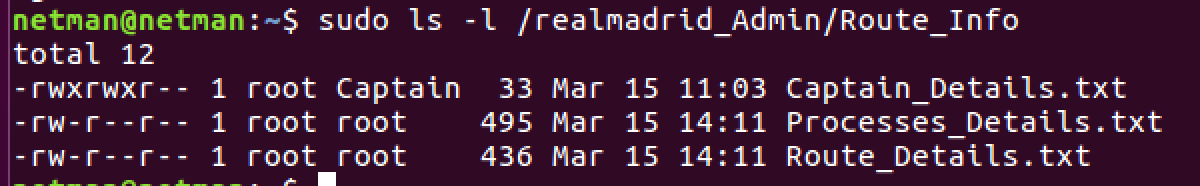
Objective 3.4 – Executing Shell Commands, Performing File Transfer

1. The FIFA\_admin machine needs to request route table and top ten running processes (based on memory usage) from the managed hosts. Add functionality in the playbook to do the following:
2. Execute shell commands (hint: netstat , ps ) to retrieve the required information and store it in two different files. **[10 Points]**









1. Perform secure transfer of these files from host to FIFA\_admin using **SCP** commands. Use the **expect** module to enter password and any other information associated with SCP when prompted. Passwords and prompts should not be handled manually but through the playbook. **[15 Points]**

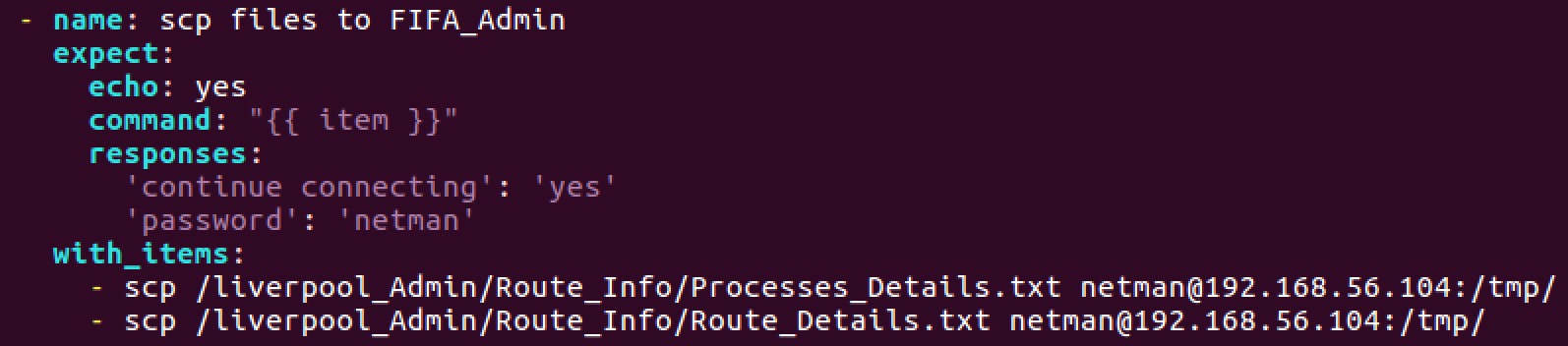
Note: You need to do the followings before doing scp in yaml

1. Install pexpect in Liverpool and realmadriod server by: **sudo -H pip2 install pexpect**
2. Change the password of root by:

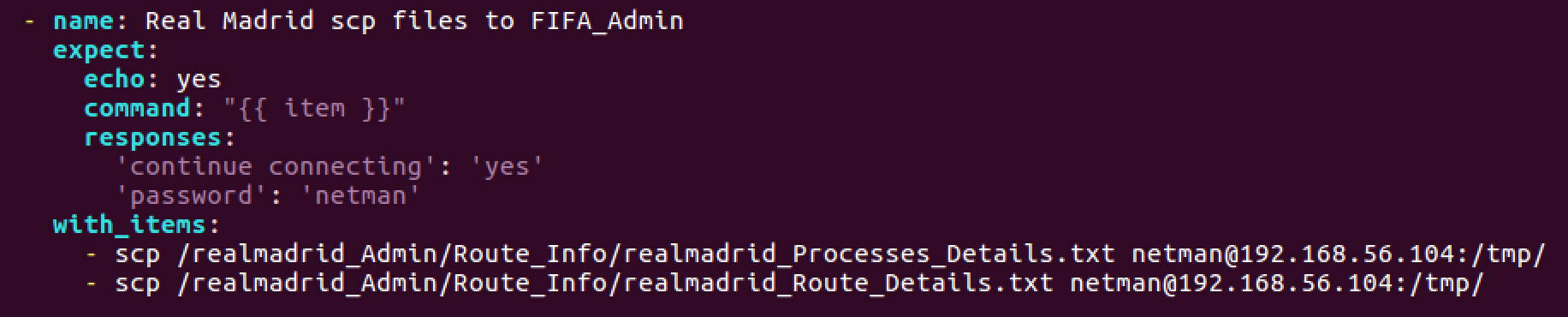
* Sudo su
* Passwd

1. Then change the ownership of Liverpool\_Admin/ and realmadrid\_Admin/ to netman, ex: sudo chown netman -R Liverpool\_Admin (This makes entire directory to be netman owner)

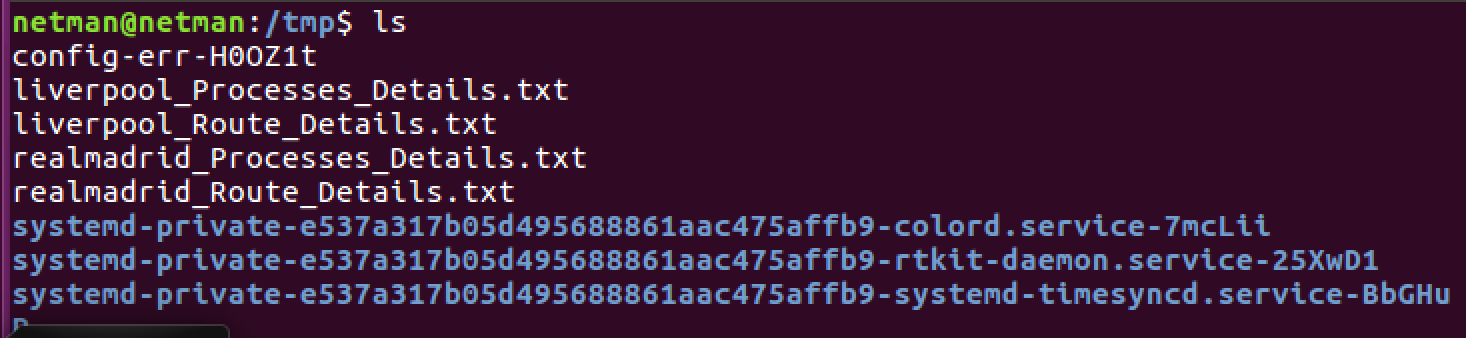
Liverpool:



Real Madriod:

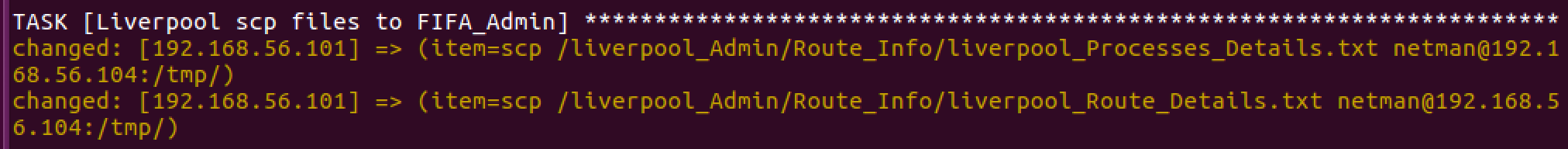


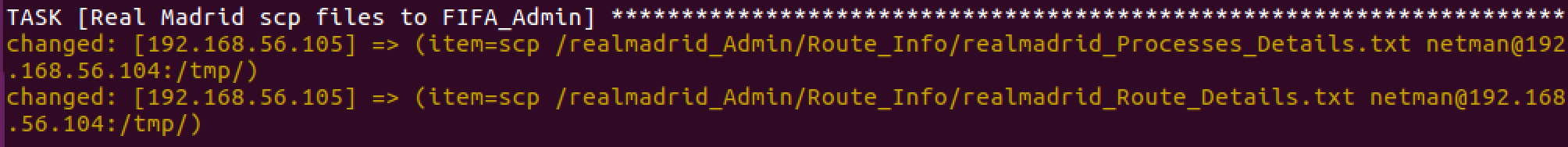
FIFA\_Admin:



1. Provide screenshots of the relevant portion of the playbook and successful file transfer.

**[5 Points]**





Objective 3.5 – Combining multiple files into a single file

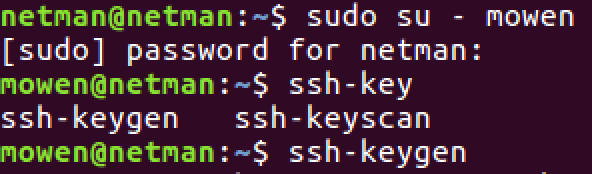
1. Manually generate SSH keys for the following players on their local machines: **[5 Points]**
   1. **Liverpool b. Real Madrid**

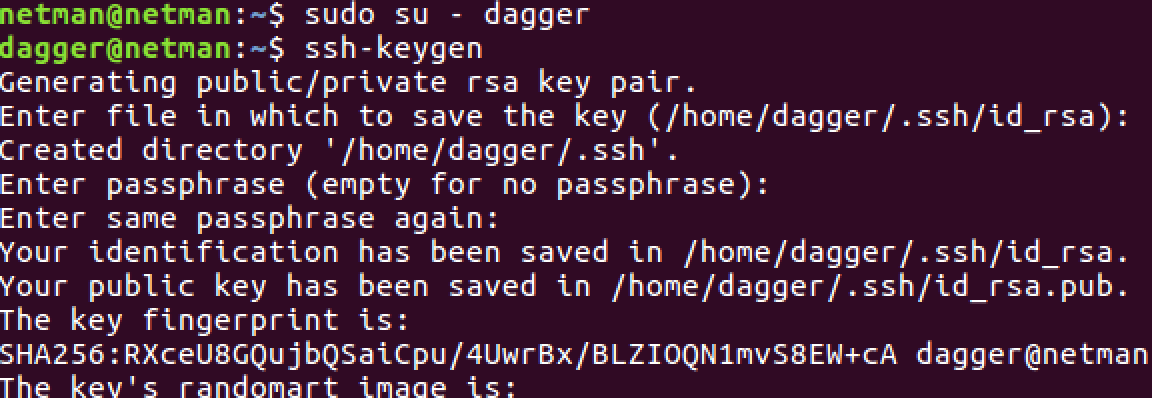
Michael Owen Luis Figo

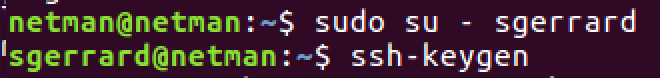
Daniel Agger Fabio Cannavaro

Steven Gerrard Zinedine Zidane

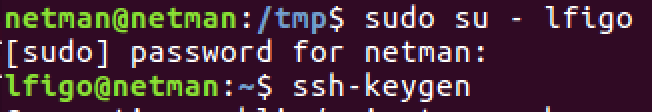
LiverPool:

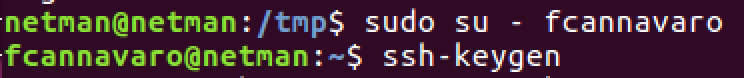


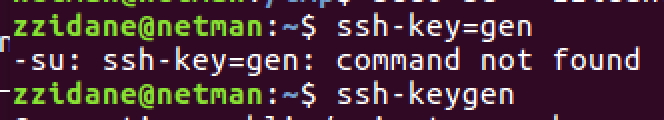




**Real Madrid:**







1. In the same playbook, use the [assemble](http://docs.ansible.com/ansible/assemble_module.html) module of Ansible to combine the public keys( .pub) for the above players into a single file on the host machine named as TeamName\_authorizedKeys.txt and store the file in the .ssh directory of the root user. (eg. Liverpool\_authorizedKeys.txt will have public keys of the three Liverpool players and will be stored in the root user’s .ssh directory on the Liverpool machine.)

**[10 Points]**

1. Using the playbook, send these files to the FIFA\_admin machine and store them in the root user’s .ssh directory. **[10 Points]**

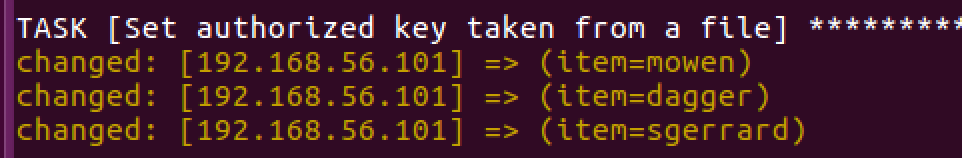
In order to do this requirement, I have to set up classVM of root/ owner is netman, not root user

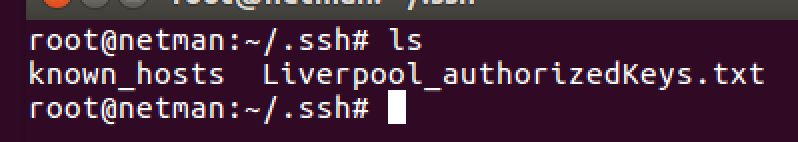


Command: sudo chown netman -R root/

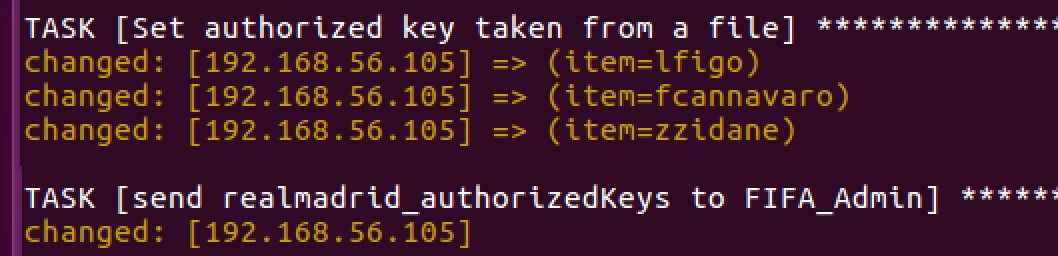
1. Provide screenshots of the relevant portion of the playbook and the files in the .ssh directory of the root user of FIFA\_admin machine. **[5 Points]**

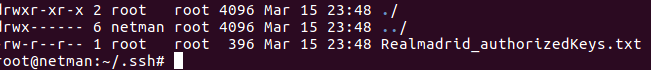
Liverpool:





Real Madrid:





Objective 4 – Additional Ansible

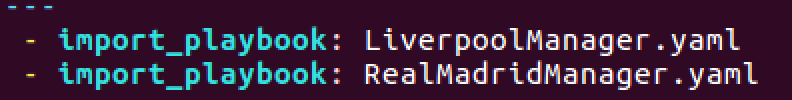
Part 1 (Using additional modules in Ansible)

1. Create two more playbooks named as LiverpoolManager.yml and RealMadridManager.yml
2. Think of a use case that will make use of the following Ansible modules in the respective YAML files: **[10 Points]**

|  |  |
| --- | --- |
| **File** | **Modules to use** |
| LiverpoolManager.yml | group\_by, wait\_for, set\_fact |
| RealMadridManager.yml | template |

1. Once you are done executing the above YAML files separately, think of executing all the above tasks via a single playbook that you created for this lab (YourIdentikey\_play.yml)

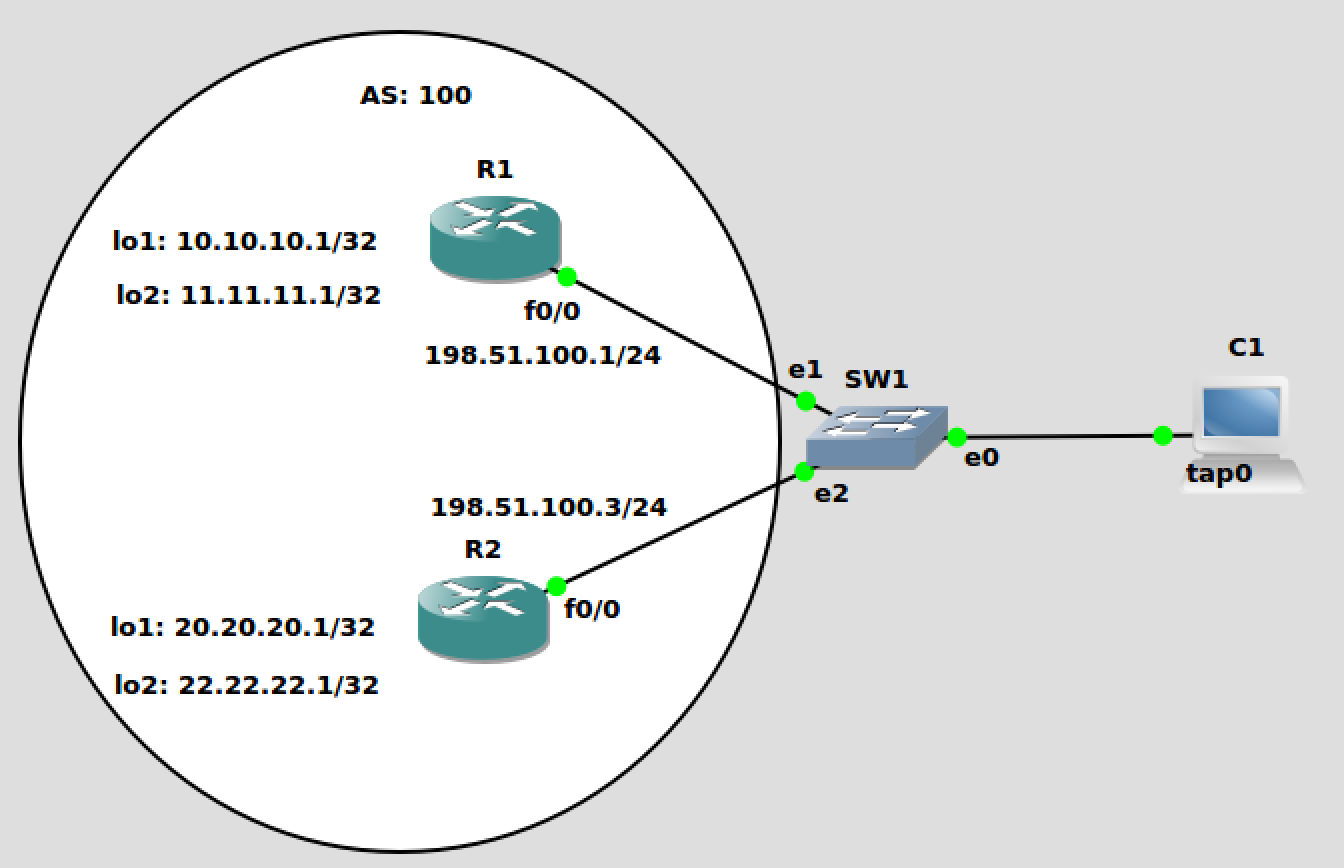
**[5 Points]**



Part 2 (Router configuration via Ansible)

1. Construct the same topology done in Automation 1 and configure IBGP between the routers by passing commands through Ansible playbook. It is not required to use multithreading in this case. **[15 Points]**

Topology:

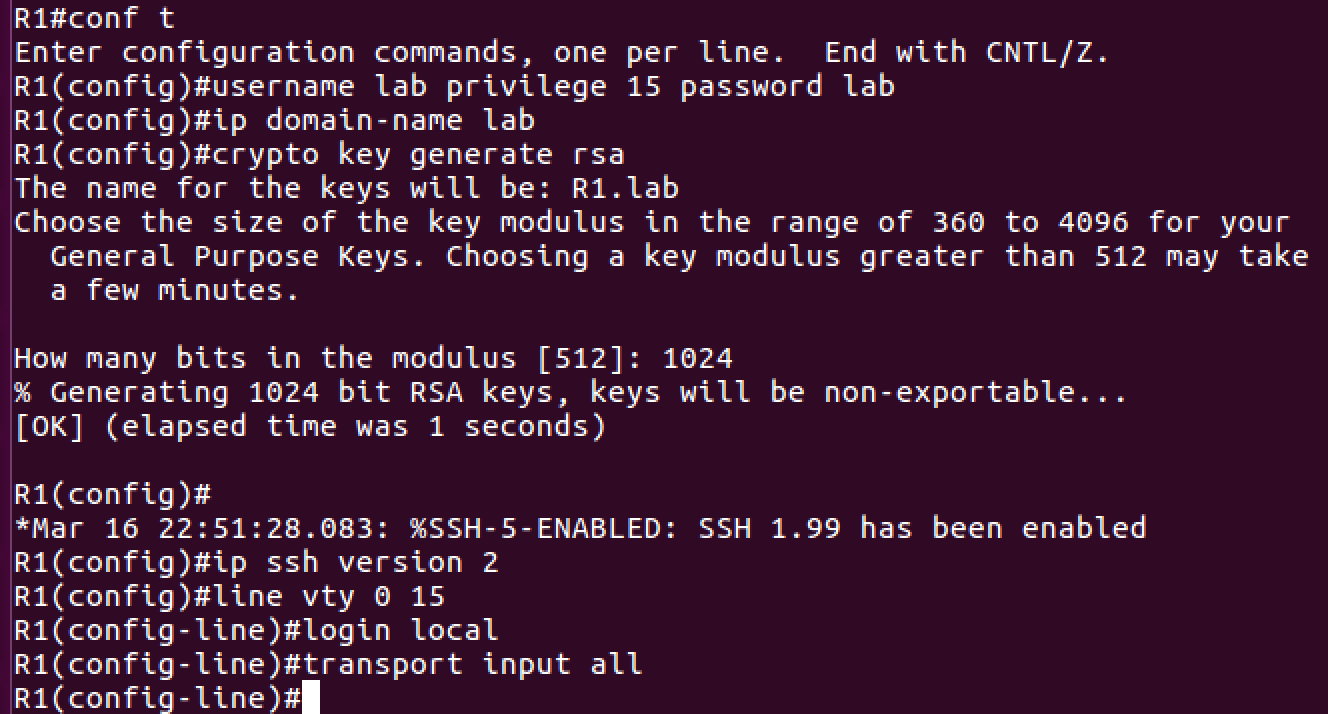


First of all, I need to set up ssh on both routers:

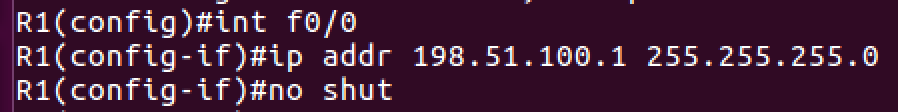
Note: because I set up ssh and I start running the command, and type the ssh password:



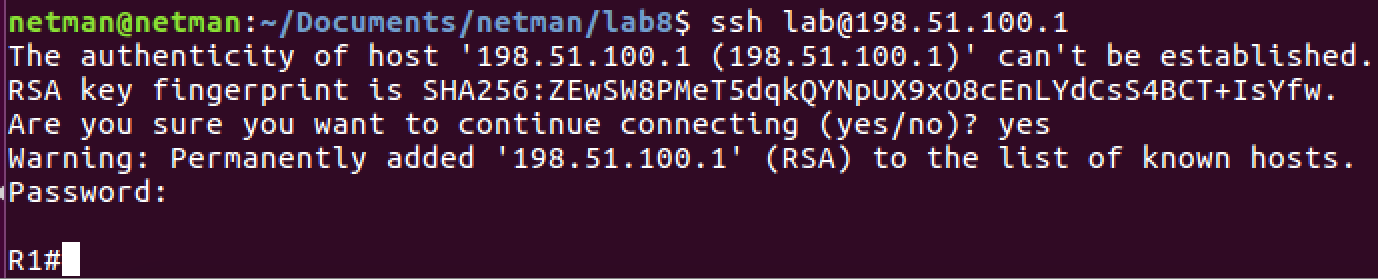
R1:



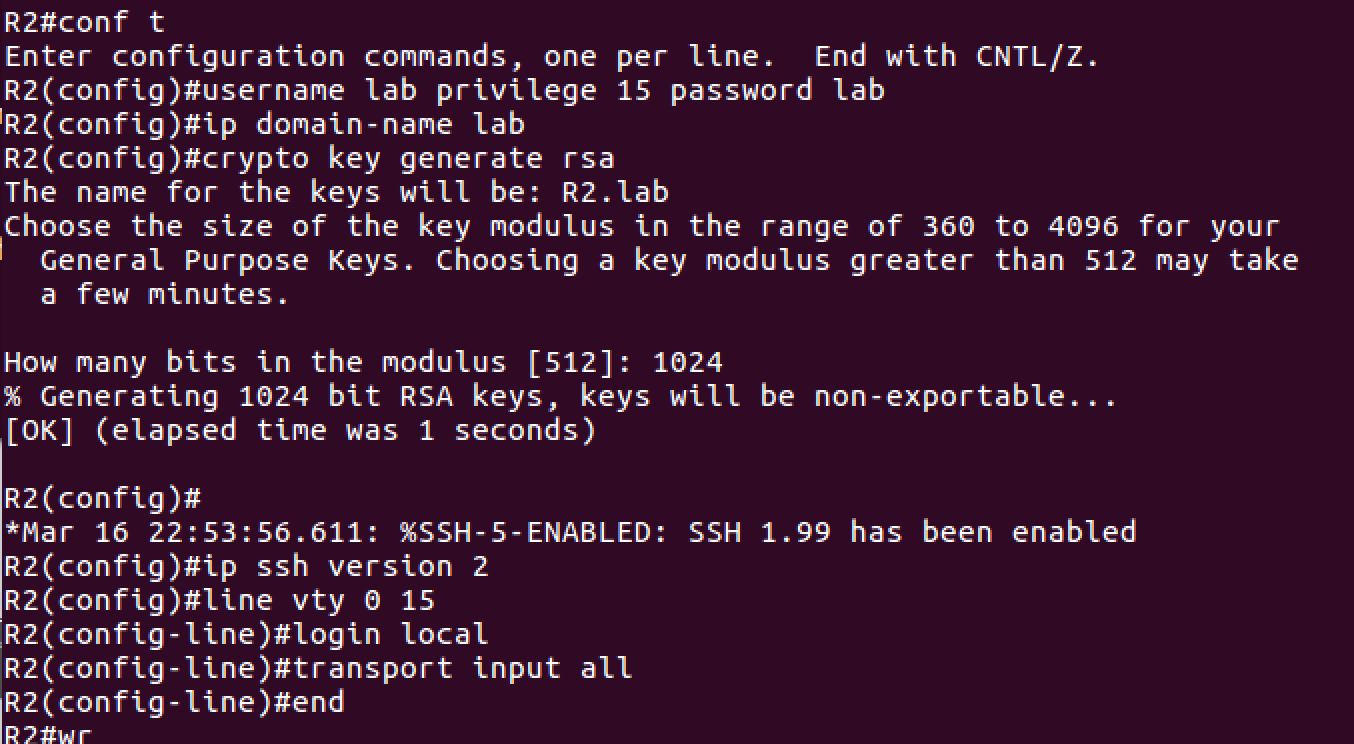
Configure interface:



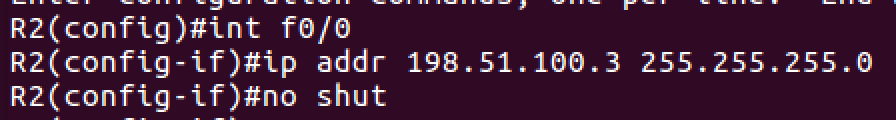
Test ssh on R1 from VM:



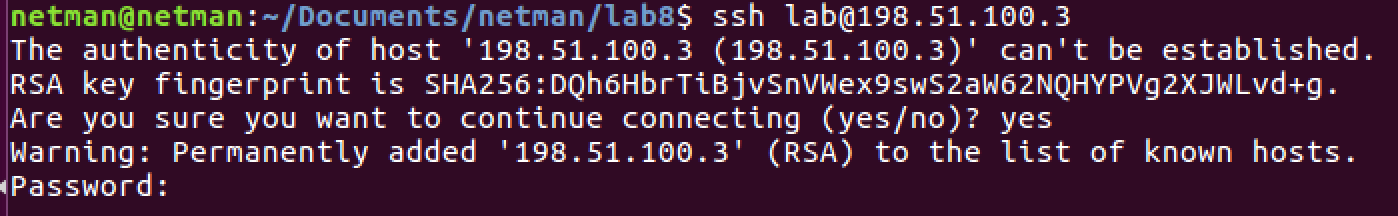
R2:



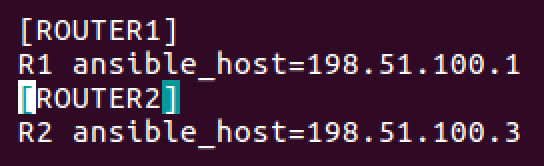
Configure interface:



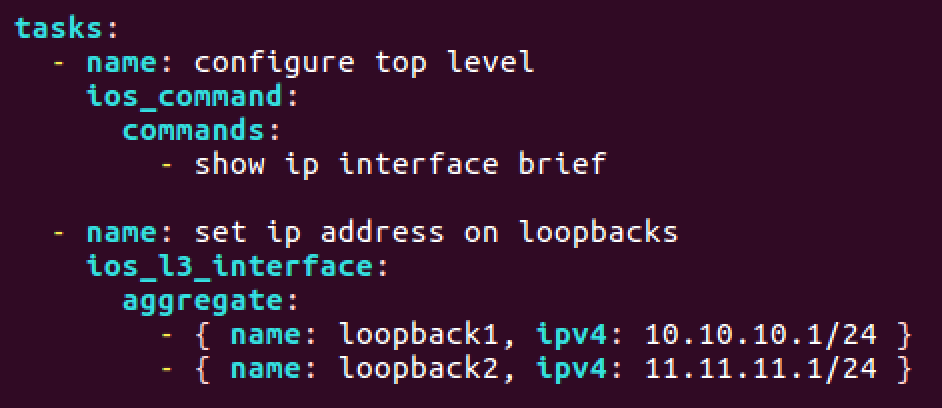
Test ssh on R2 from VM:



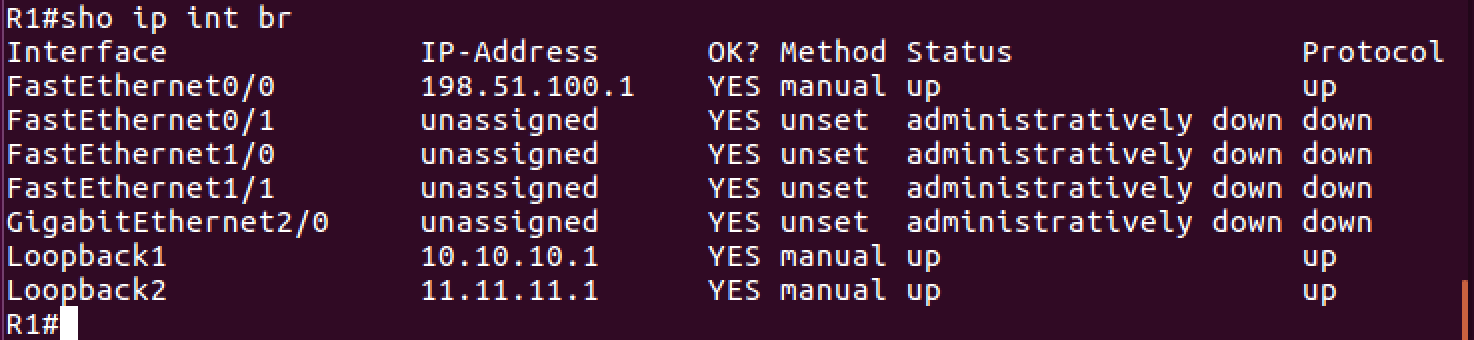
Add this to /etc/ansible/host



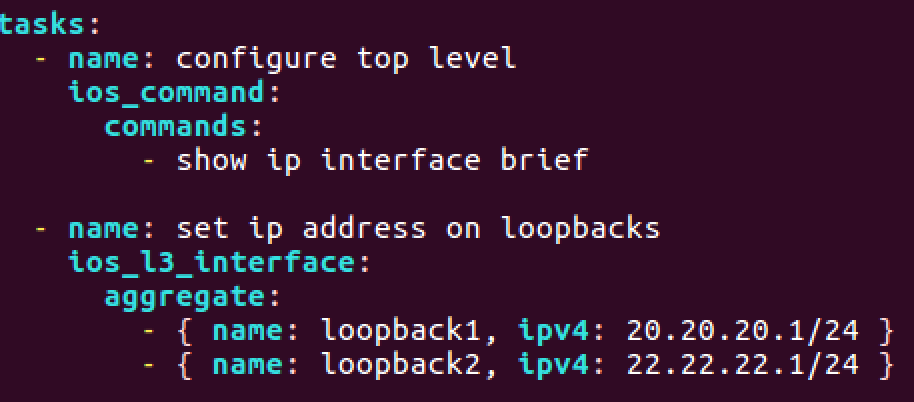
Configure loopback interfaces on R1:



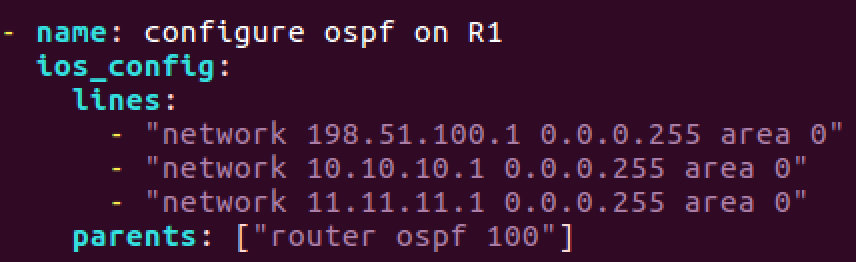
To check:

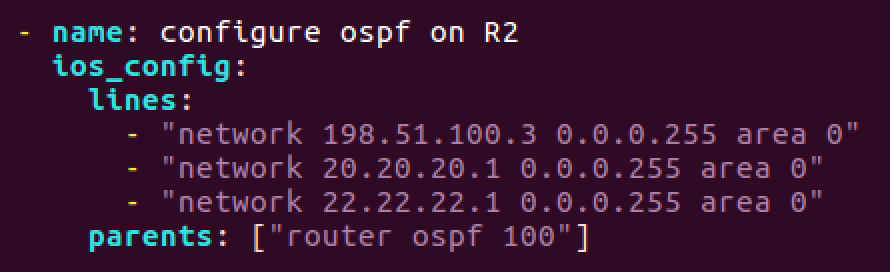


R2:

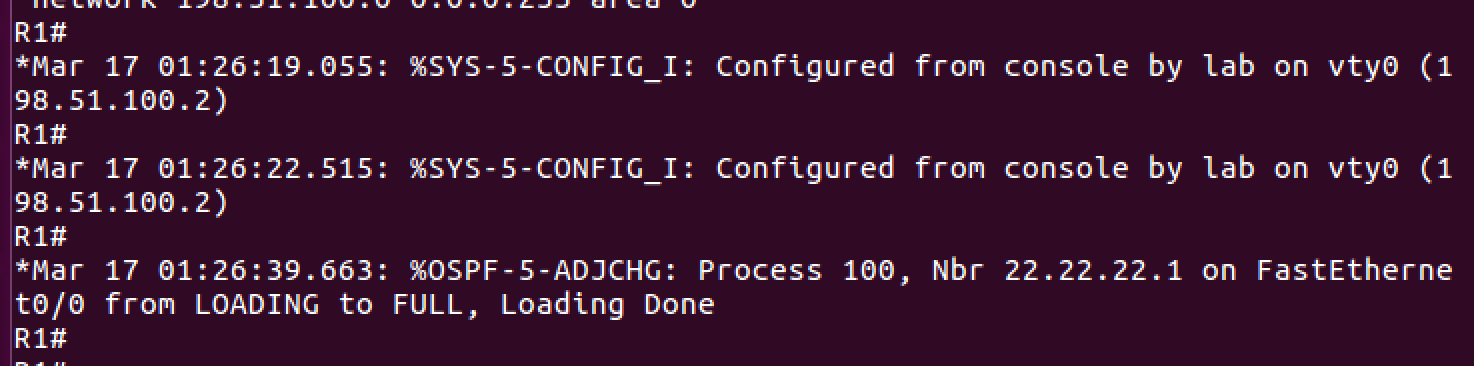


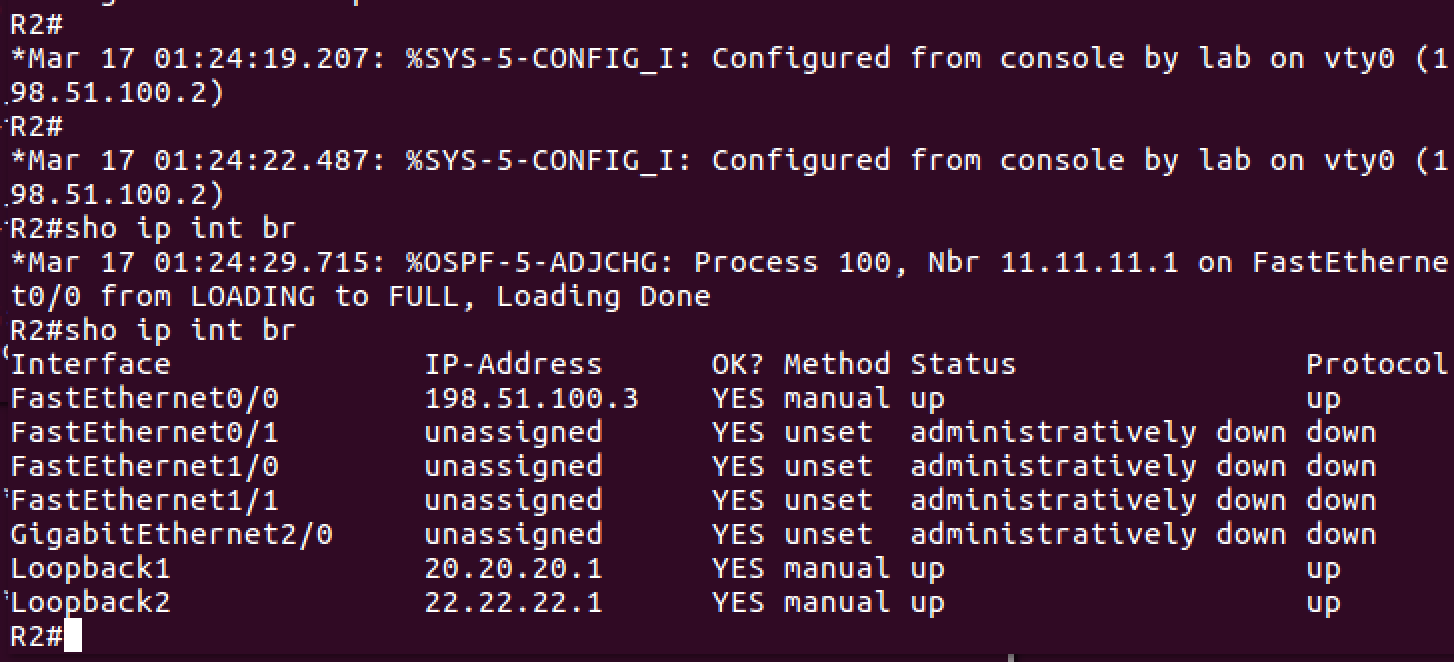
Configure OSPF on R1 and R2:



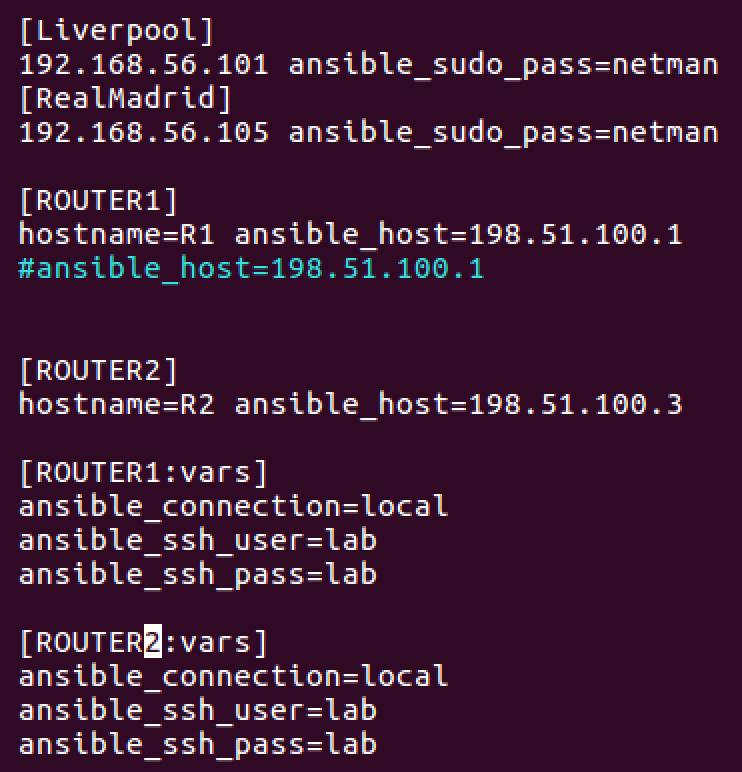


Results:





All my configuration in /etc/ansible/hosts:



Total Points \_\_\_\_\_\_\_\_\_\_\_\_ / 175