



# **PRA TEST PROJECT** **MODUL C – INFRASTRUCTURE** **PROGRAMMABLE & AUTOMATION**

## ***IT NETWORK SYSTEMS ADMINISTRATION***

**LOMBA KOMPETENSI SISWA SMK**  
**TINGKAT NASIONAL 2024**

*Dokumen ini merupakan **Pra-Test Project** yang menjadi subject **perubahan maksimal 30%** untuk Actual Test Project. Pelaksanaan kompetisi LKS Nasional nanti akan menggunakan **Actual Test Project** yang akan dipublikasi pada **saat kompetisi**. Perubahan tersebut meliputi Topology, Functionality, Aplikasi dan Task yang diminta.*

## Description of project and tasks

- We have installed these IDE in JUMPHOST.
  - VSCode
  - Atom
  - Other IDE available in DLBD can also be installed and used.
- No Offline Documentations

## Instructions to the Competitor

- We require you to use ansible playbooks, but you may also use Bash and/or Python script for task that are not inventory, with 50% score reduction for that particular task.
- Ansible playbook or script must be run by user root from JUMPHOST, and must not require any manual authentication.
- Please write command example on how to run the playbook (or script if you choose) in /etc/motd in JUMPHOST, we provided an example file in the appendix.
- If your script or playbook is not documented, we may use it wrongly and you may lose score.
- Filename is provided in each task, please place all playbook in /root/ansible.
- If you choose to use Bash/Python, place the scripts in /root/scripts, change the filename from .yml to .sh or .py, and give proper permission to run the script.
- We will restore the vm to fresh state, and then run the playbook or script to check functionality of each task.

## Part I – Basic Automation

1. Create basic inventory file in `/etc/ansible/all-linux` containing all linux hosts.
2. Create role-based inventory file that is compatible with other tasks:
3. `/etc/ansible/site-1`, with this setup:
  - Sinabung as **dns**
  - Rinjani as **web**
  - Serua as **web**
4. `/etc/ansible/site-2`, with this setup:
  - Rinjani as **dns**
  - Sinabung as **web**
5. Create playbook with name **hostname-all.yml** to change all hostname in all linux hosts to the correct as shown in Topology.
6. Create playbook with name **dns-install.yml** to install DNS service in all linux hosts. You may use any DNS service.
7. Create playbook with name **dns-config.yml** to configure DNS service in all linux hosts to listen on each IP addresses they have, **but not on 0.0.0.0**. For example, if the host has 192.168.1.1 and 127.0.0.1, then the service must listen on these IP only.

## Part II – Idempotency

1. Create playbook with name `etc-hosts-all.yml` to change `/etc/hosts` record in all linux hosts.
2. Make sure same records are not created multiple time.
3. Create playbook with name `haproxy-all.yml` to install and configure reverse proxy in port 8081, to apache service on localhost. This must be executed on all linux hosts.

## Part III – Specific Host

1. Restart Create playbook with name **restart-dns.yml** to restart hosts with **dns** role.
2. Create playbook with name **restart-web.yml** to restart hosts with **web** role.
3. Create playbook with name **stop-dns.yml** to stop DNS service in hosts with **dns** role.

## Part IV – Arguments

1. Create Create playbook with name **add-user.yml** to create user **natuna** in all linux hosts, with password Skills39.
2. Create playbook with name **install-package.yml** to install a package in all linux hosts, by reading from a textfile `/root/packages.txt`. Sample textfile is provided in the appendix, and we will change the textfile to grade the task later

## Part V – Complex

1. Create playbook with name **apache-install.yml** to install and configure apache service in all linux hosts with these details (change hostname with each hostname):
  - Edit default index.html to “Welcome to `$(hostname)`”
  - Create webpage for all users in `/root/usernames.txt`
    - `http://$(ip_address)/$(username)/index.html`
    - ...
    - `http://$(ip_address)/$(username)/index.html`

- index.html content refer to appendix.
- 2. Create playbook with name **dns-add-record.yml** to add specific DNS entry to the DNS service from textfile **/root/dnslist.txt** :
  - Automatically restart service if there is change.
  - Don't restart service if no change.
  - Apply to all hosts with condition if DNS service installed.
  - If DNS service not installed then it must not return an error but just skip the execution.

## Part VI – Windows

1. Create playbook with name **windows-iis.yml** to install IIS in windows server.
2. Create playbook with name **windows-dns.yml** to install DNS in windows hosts and create. following record:
  - A record pointing awu.lks2024.id to 10.17.8.45

## Appendix

- /etc/motd
  - ## Change hostname on all linux hosts  
ansible-playbook -i /root/ansible/all-linux /root/ansible/hostname-all.yml
  - ## Install dnsmasq in all linux hosts  
ansible-playbook -i /root/ansible/all-linux /root/ansible/dnsmasq-all.yml
  - ## Change /etc/hosts on all linux hosts  
ansible-playbook -i /root/ansible/ all-linux /root/ansible/etc-hosts-all.yml

- /root/usernames.txt

User	Password
abdul	password123
ahmad	password345
salma	passwerd111
ihsan	asdfghjkl
bagas	qwertyuiop
saepul	11223344

- /root/packages.txt
  - **links**
  - **curl**
  - **wget**
- index.html
  - Make sure variable username and hostname are rendered correctly  
“ This is homepage for \$(username) in \$(hostname) ”

# Topology

