

<b>Name: Ballesteros, John Erwin S.</b>	<b>Date Performed: 21/10/2024</b>
<b>Course/Section: CpE31s2</b>	<b>Date Submitted: 28/10/2024</b>
<b>Instructor: Engr. Robin Valenzuela</b>	<b>Semester and SY: 1st Sem, '24-'25</b>
<b>Activity 9: Install, Configure, and Manage Performance Monitoring tools</b>	
<b>1. Objectives</b>	
Create and design a workflow that installs, configure and manage enterprise performance tools using Ansible as an Infrastructure as Code (IaC) tool.	
<b>2. Discussion</b>	
<p>Performance monitoring is a type of monitoring tool that identifies current resource consumption of the workload, in this page we will discuss multiple performance monitoring tool.</p> <p><b>Prometheus</b></p> <p>Prometheus fundamentally stores all data as timeseries: streams of timestamped values belonging to the same metric and the same set of labeled dimensions. Besides stored time series, Prometheus may generate temporary derived time series as the result of queries. Source: <a href="#">Prometheus - Monitoring system &amp; time series database</a></p> <p><b>Cacti</b></p> <p>Cacti is a complete network graphing solution designed to harness the power of RRDTool's data storage and graphing functionality. Cacti provides a fast poller, advanced graph templating, multiple data acquisition methods, and user management features out of the box. All of this is wrapped in an intuitive, easy to use interface that makes sense for LAN-sized installations up to complex networks with thousands of devices. Source: <a href="#">Cacti® - The Complete RRDTool-based Graphing Solution</a></p>	
<b>3. Tasks</b>	
<ol style="list-style-type: none"> <li>1. Create a playbook that installs Prometheus in both Ubuntu and CentOS. Apply the concept of creating roles.</li> <li>2. Describe how you did step 1. (Provide screenshots and explanations in your report. Make your report detailed such that it will look like a manual.)</li> <li>3. Show an output of the installed Prometheus for both Ubuntu and CentOS.</li> <li>4. Make sure to create a new repository in GitHub for this activity.</li> </ol>	
<b>4. Output (screenshots and explanations)</b>	
<p><b>Step 1:</b></p> <p><b>Clone your created repository</b></p>	

## Step 2:

Create the appropriate directory for the roles and tasks, where inside each roles there are directory named “tasks” inside



## Step 3:

In the main directory create a file named run.yml with the following code:

```
19
20   - hosts: all
21     become: true
22     roles:
23       - base
24
25   - hosts: workstations
26     become: true
27     roles:
28       - workstations
29
30   - hosts: web_servers
31     become: true
32     roles:
33       - web_servers
34
35   - hosts: db_servers
36     become: true
37     roles:
38       - db_servers
39
40   - hosts: file_servers
41     become: true
42     roles:
43       - file_servers
```

#### Step 4: Create the main.yml with the following code:

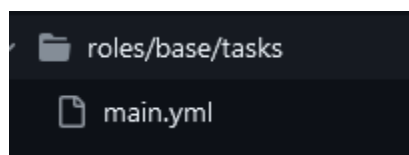
```
1  ---
2  - name: install prometheus
3    tags: install
4    get_url:
5      url: https://github.com/prometheus/prometheus/releases/download/v2.54.1/prometheus-2.54.1.linux-amd64.tar.gz
6      dest: /tmp/prometheus.tar.gz
7
8  - name: extract prometheus
9    tags: extract
10   unarchive:
11     src: /tmp/prometheus.tar.gz
12     dest: /usr/local/bin
13     remote_src: yes
14
15  - name: create prometheus group
16    user:
17      name: prometheus
18      shell: /sbin/nologin
19
20  - name: copy prometheus
21    copy:
22      src: /usr/local/bin/prometheus-2.54.1.linux-amd64/{{ item }}
23      dest: /usr/local/bin
24      mode: '0755'
25    with_items:
26      - prometheus
27      - promtool
28
29  - name: create config dir prometheus
30    file:
31      path: /etc/prometheus
32      state: directory
33      owner: prometheus
34      group: prometheus
35
```

```

35
36   - name: copy prometheus config
37     copy:
38       src: /usr/local/bin/prometheus-2.54.1.linux-amd64/prometheus.yml
39       dest: /etc/prometheus/prometheus.yml
40       owner: prometheus
41       group: prometheus
42       mode: '0644'
43
44   - name: set up prometheus systemd service
45     copy:
46       dest: /etc/systemd/system/prometheus.service
47       content: |
48         [Unit]
49         Description=Prometheus
50         After=network.target
51
52         [Service]
53         User=prometheus
54         ExecStart=/usr/local/bin/prometheus \
55           --config.file=/etc/prometheus/prometheus.yml \
56           --storage.tsdb.path=/var/lib/prometheus
57         Restart=always
58
59         [Install]
60         WantedBy=multi-user.target
61
62   - name: reload systemd and enable prometheus
63     systemd:
64       daemon_reload: yes
65       name: prometheus
66       enabled: yes
67       state: started

```

**Step 5: Copy the main.yml into the base/tasks directory. You can also copy this individually on each server and not in the base. Using the base/tasks directory allows us to install it on all of the servers**



**Step 6: Run the playbook using “ansible-playbook –ask-become-pass run.yml”**

**Result of running the Playbook:**

```
TASK [base : install prometheus] *****
ok: [192.168.56.110]
ok: [192.168.56.113]
ok: [192.168.56.114]
ok: [192.168.56.109]

TASK [base : extract prometheus] *****
ok: [192.168.56.113]
ok: [192.168.56.110]
ok: [192.168.56.109]
ok: [192.168.56.114]

TASK [base : create prometheus group] *****
ok: [192.168.56.110]
ok: [192.168.56.114]
ok: [192.168.56.109]
ok: [192.168.56.113]

TASK [base : copy prometheus] *****
ok: [192.168.56.109] => (item=prometheus)
ok: [192.168.56.114] => (item=prometheus)
ok: [192.168.56.110] => (item=prometheus)
ok: [192.168.56.113] => (item=prometheus)
ok: [192.168.56.109] => (item=promtool)
ok: [192.168.56.114] => (item=promtool)
ok: [192.168.56.110] => (item=promtool)
ok: [192.168.56.113] => (item=promtool)

TASK [base : create config dir prometheus] *****
ok: [192.168.56.114]
```

```
TASK [base : create config dir prometheus] *****
ok: [192.168.56.114]
ok: [192.168.56.109]
ok: [192.168.56.110]
ok: [192.168.56.113]

TASK [base : copy prometheus config] *****
ok: [192.168.56.114]
ok: [192.168.56.109]
ok: [192.168.56.110]
ok: [192.168.56.113]

TASK [base : set up prometheus systemd service] *****
ok: [192.168.56.109]
ok: [192.168.56.114]
ok: [192.168.56.110]
changed: [192.168.56.113]

TASK [base : reload systemd and enable prometheus] *****
changed: [192.168.56.110]
changed: [192.168.56.113]
changed: [192.168.56.109]
changed: [192.168.56.114]
```

### Checking if prometheus was installed

```
erwin@workstation:~/Act9_Ballesteros$ prometheus --version
prometheus, version 2.54.1 (branch: HEAD, revision: e6cfa720fbe6280153fab13090a483dbd40bece3)
  build user:      root@812ffd741951
  build date:      20240827-10:56:41
  go version:      go1.22.6
  platform:        linux/amd64
  tags:            netgo,builtinassets,stringlabels
```

```
erwin@Server1:~$ prometheus --version
prometheus, version 2.54.1 (branch: HEAD, revision: e6cfa720fbe6280153fab13090a483dbd40bece3)
  build user:      root@812ffd741951
  build date:      20240827-10:56:41
  go version:      go1.22.6
  platform:        linux/amd64
  tags:            netgo,builtinassets,stringlabels
erwin@Server1:~$
```

```
erwin@Server2:~$ prometheus --version
prometheus, version 2.54.1 (branch: HEAD, revision: e6cfa720fbe6280153fab13090a483dbd40bece3)
  build user:      root@812ffd741951
  build date:      20240827-10:56:41
  go version:      go1.22.6
  platform:        linux/amd64
  tags:            netgo,builtinassets,stringlabels
erwin@Server2:~$
```

```
[erwin@centos9 ~]$ prometheus --version
prometheus, version 2.54.1 (branch: HEAD, revision: e6cfa720fbe6280153fab13090a483dbd40bece3)
  build user:      root@812ffd741951
  build date:      20240827-10:56:41
  go version:      go1.22.6
  platform:        linux/amd64
  tags:            netgo,builtinassets,stringlabels
[erwin@centos9 ~]$
```

## Reflections:

### Answer the following:

#### 1. What are the benefits of having a performance monitoring tool?

There are many benefits of implementing a performance monitoring tool at an industry level because it allows us to detect early issues that are not commonly detected by the average person. It also helps us by analyzing the status and it can allow us to specify the root error. By implementing this it helps in the reduction of downtimes and faster response times by allowing us to prevent the error that will happen. Lastly it is energy efficient and can help the company reduce expenses.

## Conclusions:

For this activity we have performed the creation of an ansible playbook with the installation of a prometheus monitoring tool. It is done by utilizing roles for each server and the use of acquiring the download link, extraction of files and various set up to be able to install the software properly for both Ubuntu and CentOS workstations.

Github Link: [Moznaim/Act9\\_Ballesteros \(github.com\)](https://github.com/Moznaim/Act9_Ballesteros)