**Installation**

The following installations are for Linux servers.

**Step 1:**

Node Exporter installation can be performed by following the steps in the link below.

<https://devopscube.com/monitor-linux-servers-prometheus-node-exporter/>

While following the steps in the link above. Under the title of “**Create a Custom Node Exporter Service**” below content will be added. **\*location\_of\_directory\*** must point the file where **\*.prom** files to be saved. Thus, Node Exporter will display the files contents at ‘**\*node\_exporter\_ip\*:9100/metrics**’

**[Unit]**  
Description=Node Exporter  
After=network.target  
  
**[Service]**  
User=node\_exporter  
Group=node\_exporter  
Type=simple  
ExecStart=/usr/local/bin/node\_exporter --collector.textfile.directory="/\*location\_of\_directory\*"  
  
**[Install]**  
WantedBy=multi-user.target

**Step 2:**

Prometheus installation can be performed by following the steps in the link below.

<https://devopscube.com/install-configure-prometheus-linux/>

While following the steps in the link above. Under the title of “**Setup Prometheus Configuration**” below content will be added. In the section of **\*node\_exporter\_ip\*** must be entered the IP address where Node Exporter installed. When the Prometheus is activated, it starts to collect and store the metrics in real time.

**global**:  
 **scrape\_interval**: 10s  
  
**scrape\_configs**:  
 - **job\_name**: 'prometheus'  
 **scrape\_interval**: 5s  
 **static\_configs**:  
 - **targets**: ['localhost:9090']  
  
 - **job\_name**: 'node\_exporter\_metrics'  
 **scrape\_interval**: 5s  
 **static\_configs**:  
 - **targets**: ['\*node\_exporter\_ip\*:9100']

**Step 3:**

Grafana installation can be performed by following the steps in the link below.

<https://devopscube.com/integrate-visualize-prometheus-grafana/>

By following the steps below, you can add prepared dashboard and you can edit the dashboard as you like. For every query you should edit **\*localhost\*** with Node Exporter’s IP and **\*node\_name\*** with the node name that you named via **get\_of\_metrics** script.

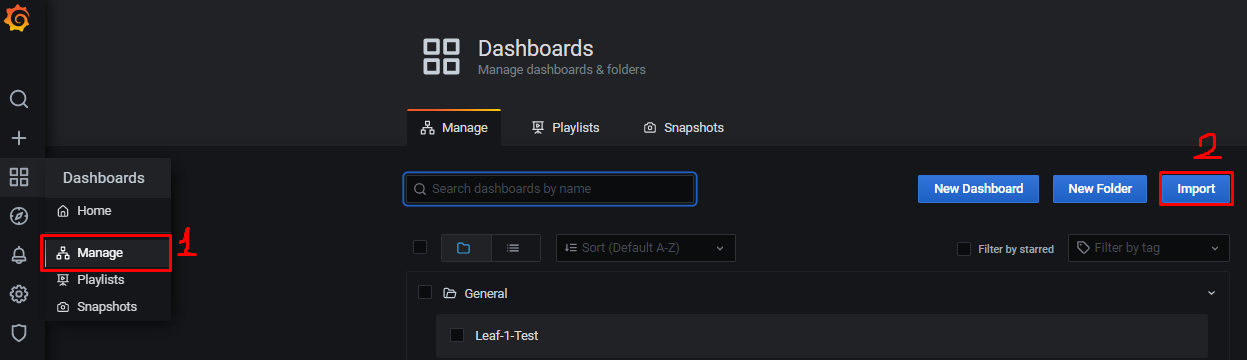
Note: node\_name is host\_name which is required by the get\_of\_metrics script.

**instance=\"\*localhost\*:9100\"” and “node\_name=\"\*node\_name\*\"**

The JSON file is at sample-dashboard.json

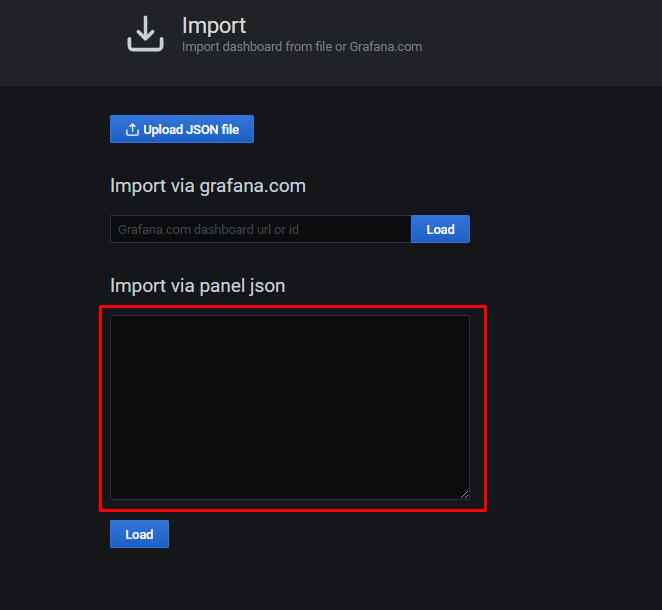
**Step 1:**

This way you can access the importing the JSON



**Step 2:**

After copying the JSON file into the box or uploading the JSON file, the dashboard installation will be done when press the "Load" button.



**Step 4:**

Installing the get\_of\_metrics.py script as a service.

**Step 1:**

Copy the get\_of\_metrics.dep file to server which will run the script constantly. So, we can install the debian package using the dpkg or apt package management tool.

**Step 2:**

**sudo apt install ./get\_of\_metrics.deb**

or

**sudo dpkg -i ./get\_of\_metrics.deb**

**Step 3:**

You need to enter the remote machine information in the order shown in the connection-parameters.json file. When you install the Dockerfile it will access the remote machine you entered in the file.

{

"port": "8000",  
 "delay": "5",  
 "hosts":  
 [  
 {  
 "alias": "Leaf-1",  
 "host": "10.10.0.31",  
 "user": "root",  
 "password": "onl"  
 },  
 {  
 "alias": "Leaf-2",  
 "host": "10.10.0.32",  
 "user": "root",  
 "password": "onl"  
 },  
 {  
 "alias": "Leaf-3",  
 "host": "10.10.0.33",  
 "user": "root",  
 "password": "onl"  
 },  
 {  
 "alias": "Leaf-4",  
 "host": "10.10.0.34",  
 "user": "root",  
 "password": "onl"  
 }  
 ]  
}