2.
Install node exporter on another machine than the server.

- **Add that machine target to server configuration.**

Answer:

I tried to use a node-exporter in another VM but due to some issue could not run both the VMs at the same time due to which state was seen down in the prometheus dashboard.

So, I used windows exporter also which is given below;

For node_exporter:

First, we install **node-exporter** as follows;

- **curl -LO https://github.com/prometheus/node_exporter/releases/download/v1.3.0/node_exporter-1.3.0.linux-amd64.tar.gz**

```
aashish@vm2: ~/node_exporter-1.3.0.linux-amd64
aashish@vm2:~$ curl -LO https://github.com/prometheus/node_exporter/releases/do
wnload/v1.3.0/node_exporter-1.3.0.linux-amd64.tar.gz
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   673  100   673    0     0    950      0 --:--:-- --:--:-- --:--:--   949
100 8818k  100 8818k    0     0  1452k      0  0:00:06  0:00:06 --:--:-- 1789k
```

Once downloaded, we unzip the downloaded file as follows;

- **tar xvfz node_exporter-1.3.0.linux.amd64.tar.gz**

```
aashish@vm2:~$ tar xvfz node_exporter-1.3.0.linux-amd64.tar.gz
node_exporter-1.3.0.linux-amd64/
node_exporter-1.3.0.linux-amd64/LICENSE
node_exporter-1.3.0.linux-amd64/NOTICE
node_exporter-1.3.0.linux-amd64/node_exporter
```

Next, we run the **node_exporter** being inside the **node_exporter-1.3.0.linux-amd64** directory as follows;

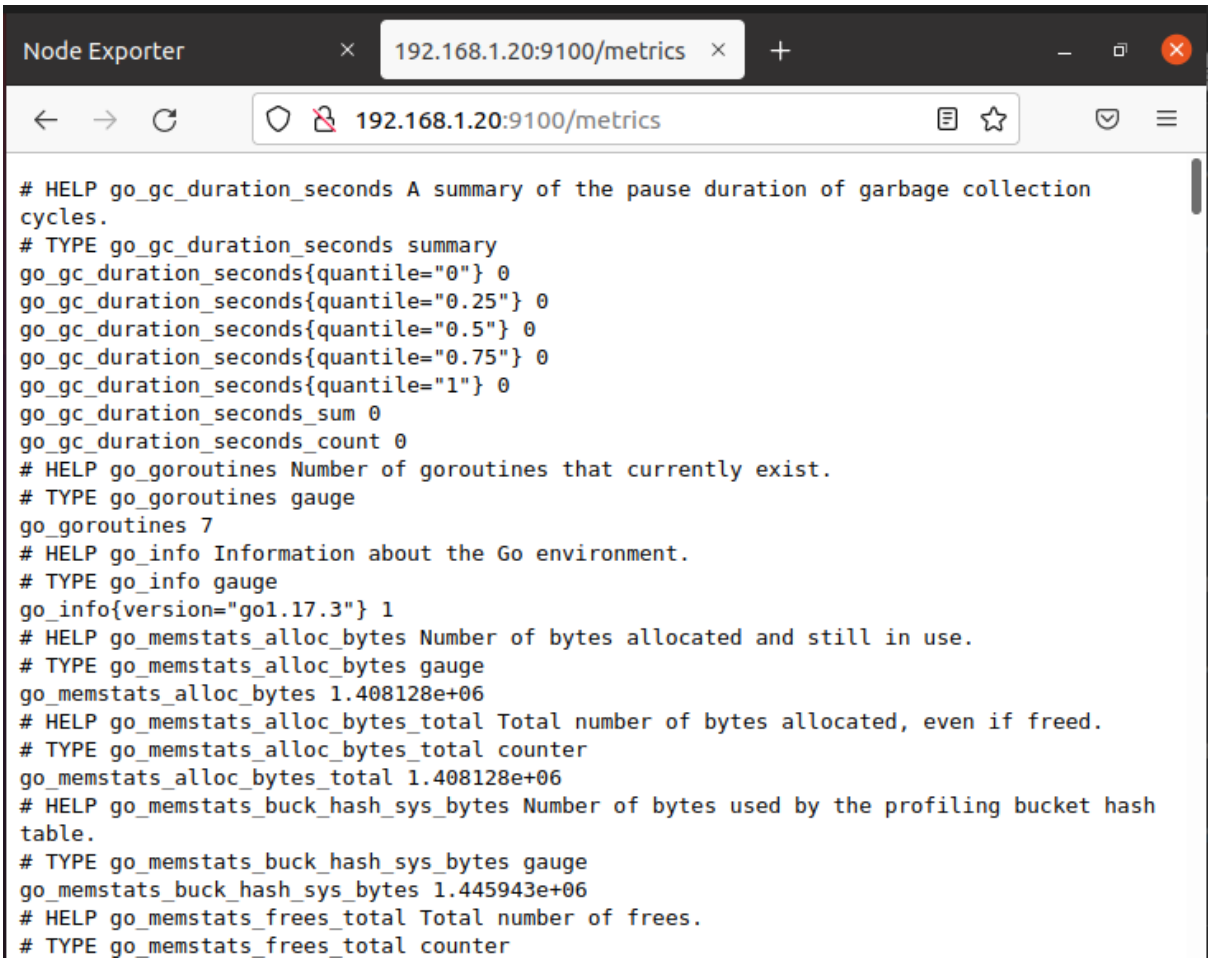
- **cd node_exporter-1.3.0.linux-amd64/**
- **./node_exporter**

```
aashish@vm2: ~/node_exporter-1.3.0.linux-amd64
aashish@vm2:~$ cd node_exporter-1.3.0.linux-amd64/
aashish@vm2:~/node_exporter-1.3.0.linux-amd64$ ./node_exporter
ts=2021-12-04T20:51:13.346Z caller=node_exporter.go:182 level=info msg="Startin
g node_exporter" version="(version=1.3.0, branch=HEAD, revision=c65f870ef90a408
8210af0319498b832360c3366)"
ts=2021-12-04T20:51:13.346Z caller=node_exporter.go:183 level=info msg="Build c
ontext" build_context="(go=go1.17.3, user=root@4801f8f250a6, date=20211118-16:3
4:14)"
ts=2021-12-04T20:51:13.347Z caller=filesystem_common.go:111 level=info collecto
r=filesystem msg="Parsed flag --collector.filesystem.mount-points-exclude" flag
=^/(dev|proc|run/credentials/.+|sys|var/lib/docker/.+)($|/)
ts=2021-12-04T20:51:13.347Z caller=filesystem_common.go:113 level=info collecto
r=filesystem msg="Parsed flag --collector.filesystem.fs-types-exclude" flag=^(a
utofs|binfmt_misc|bpf|cgroup2?|configfs|debugfs|devpts|devtmpfs|fusectl|hugetlb
fs|iso9660|mqueue|nsfs|overlay|proc|procfs|pstore|rpc_pipefs|securityfs|selinux
fs|squashfs|sysfs|tracefs)$
ts=2021-12-04T20:51:13.350Z caller=node_exporter.go:108 level=info msg="Enabled
 collectors"
ts=2021-12-04T20:51:13.351Z caller=node_exporter.go:115 level=info collector=ar
p
ts=2021-12-04T20:51:13.351Z caller=node_exporter.go:115 level=info collector=bc
ache
ts=2021-12-04T20:51:13.351Z caller=node_exporter.go:115 level=info collector=bo
nding
ts=2021-12-04T20:51:13.351Z caller=node_exporter.go:115 level=info collector=bt
rfs
ts=2021-12-04T20:51:13.351Z caller=node_exporter.go:115 level=info collector=co
nntrack
ts=2021-12-04T20:51:13.351Z caller=node_exporter.go:115 level=info collector=cp
```

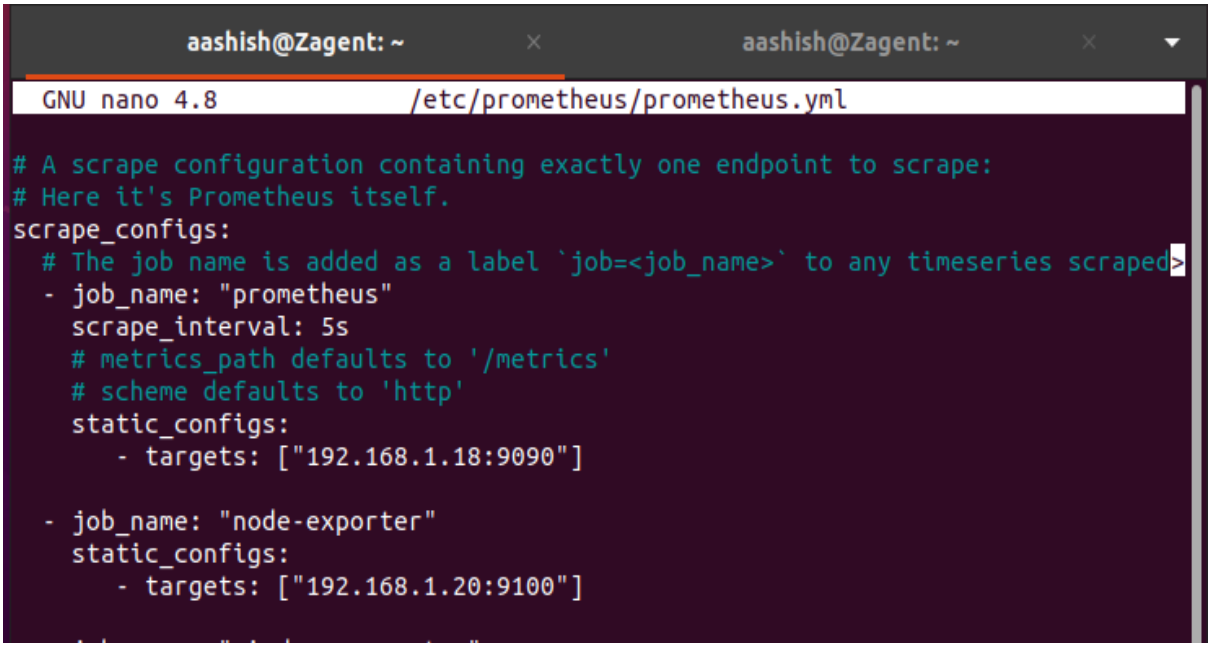
Next, we verify it via web browser using the ip(VMs IP) as follows;

- **http://192.168.1.20:9100**

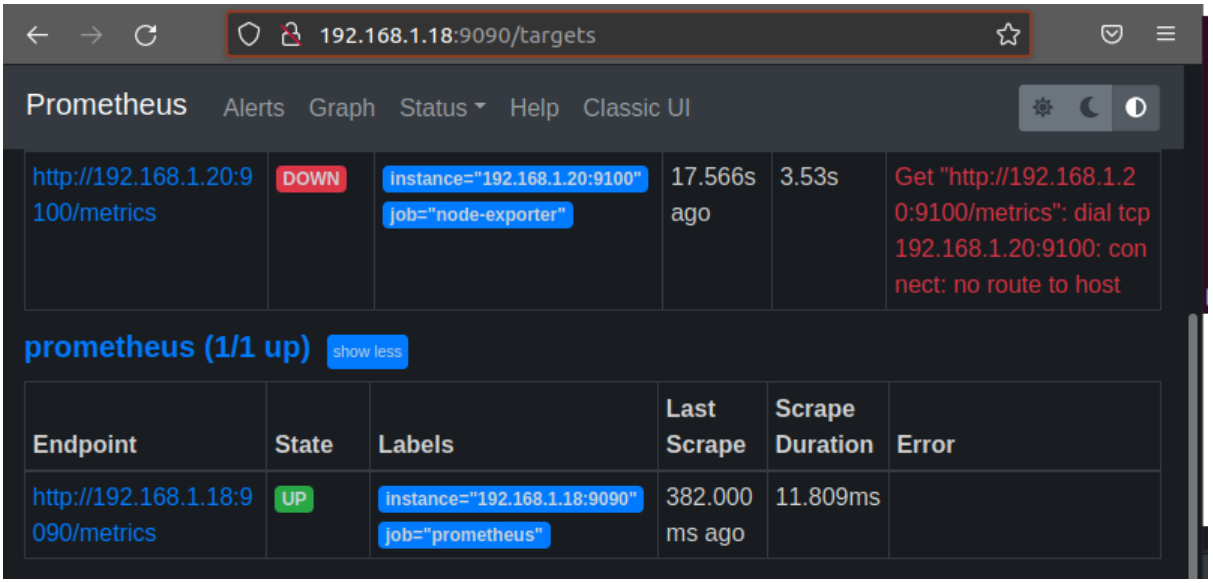
We click on the metrics to see the metrics as below;



Next, we edit **prometheus.yml** as follows;



We click on the **status -> targets,** to check the targets as follows;



Could not run both VMs at a time so, it is shown down on the targets.
So, I have used a Windows exporter.

For windows_exporter:

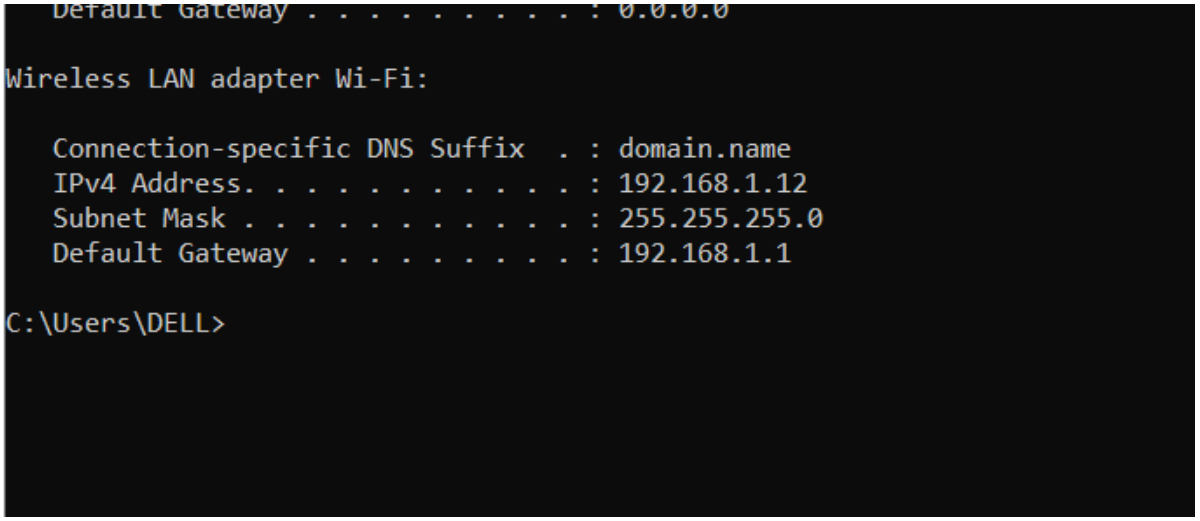
To download the windows_exporter I used the following repository;

- **https://github.com/prometheus-community/windows_exporter/releases**

And downloaded the exporter msi file as follows;

- **windows_exporter-0.16.0-amd64.msi**

Next, we check the IP address of the host machine as follows;



```
   Default Gateway . . . . . . . . . : 0.0.0.0

Wireless LAN adapter Wi-Fi:

   Connection-specific DNS Suffix  . : domain.name
   IPv4 Address. . . . . . . . . . . : 192.168.1.12
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.1.1

C:\Users\DELL>
```

Next, we edit prometheus.yml as follows;

```
aashish@Zagent: ~                    aashish@Zagent: ~
  GNU nano 4.8                /etc/prometheus/prometheus.yml

# A scrape configuration containing exactly one endpoint to scrape:
# Here it's Prometheus itself.
scrape_configs:
  # The job name is added as a label `job=<job_name>` to any timeseries scraped>
  - job_name: "prometheus"
    scrape_interval: 5s
    # metrics_path defaults to '/metrics'
    # scheme defaults to 'http'
    static_configs:
       - targets: ["192.168.1.18:9090"]

  - job_name: "node-exporter"
    static_configs:
       - targets: ["192.168.1.20:9100"]

  - job_name: "windows-exporter"
    static_configs:
       - targets: ["192.168.1.12:9182"]


^G Get Help   ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify    ^C Cur Pos
^X Exit       ^R Read File  ^\ Replace    ^U Paste Text ^T To Spell   ^_ Go To Line
```

Then, we check the windows_exporter via web browser as follows;

Not secure | 192.168.1.12:9182

Apps YouTube Maps TD-W9970 DevOps handy tools

windows_exporter

Metrics

(version=0.16.0, branch=master, revision=f316d81d50738eb0410b0748c5dcdc6874afe95a)

To check wondows_exporter metrics, we click on the metric as follows;

Not secure | 192.168.1.12:9182/metrics

Apps YouTube Maps TD-W9970 DevOps handy tools

```
# HELP go_gc_duration_seconds A summary of the pause duration of garbage collection cycles.
# TYPE go_gc_duration_seconds summary
go_gc_duration_seconds{quantile="0"} 0
go_gc_duration_seconds{quantile="0.25"} 0
go_gc_duration_seconds{quantile="0.5"} 0
go_gc_duration_seconds{quantile="0.75"} 0
go_gc_duration_seconds{quantile="1"} 0
go_gc_duration_seconds_sum 0
go_gc_duration_seconds_count 14
# HELP go_goroutines Number of goroutines that currently exist.
# TYPE go_goroutines gauge
go_goroutines 14
# HELP go_info Information about the Go environment.
# TYPE go_info gauge
go_info{version="go1.15.6"} 1
# HELP go_memstats_alloc_bytes Number of bytes allocated and still in use.
# TYPE go_memstats_alloc_bytes gauge
go_memstats_alloc_bytes 4.460552e+06
# HELP go_memstats_alloc_bytes_total Total number of bytes allocated, even if freed.
# TYPE go_memstats_alloc_bytes_total counter
go_memstats_alloc_bytes_total 4.4578728e+07
# HELP go_memstats_buck_hash_sys_bytes Number of bytes used by the profiling bucket hash table.
# TYPE go_memstats_buck_hash_sys_bytes gauge
go_memstats_buck_hash_sys_bytes 1.455185e+06
# HELP go_memstats_frees_total Total number of frees.
# TYPE go_memstats_frees_total counter
go_memstats_frees_total 63199
# HELP go_memstats_gc_cpu_fraction The fraction of this program's available CPU time used by the GC since the program started.
# TYPE go_memstats_gc_cpu_fraction gauge
go_memstats_gc_cpu_fraction 0.004054583726995475
# HELP go_memstats_gc_sys_bytes Number of bytes used for garbage collection system metadata.
# TYPE go_memstats_gc_sys_bytes gauge
go_memstats_gc_sys_bytes 3.311944e+06
# HELP go_memstats_heap_alloc_bytes Number of heap bytes allocated and still in use.
# TYPE go_memstats_heap_alloc_bytes gauge
go_memstats_heap_alloc_bytes 4.460552e+06
# HELP go_memstats_heap_idle_bytes Number of heap bytes waiting to be used.
# TYPE go_memstats_heap_idle_bytes gauge
go_memstats_heap_idle_bytes 5.865472e+06
# HELP go_memstats_heap_inuse_bytes Number of heap bytes that are in use.
# TYPE go_memstats_heap_inuse_bytes gauge
go_memstats_heap_inuse_bytes 6.455296e+06
# HELP go_memstats_heap_objects Number of allocated objects.
# TYPE go_memstats_heap_objects gauge
go_memstats_heap_objects 27665
# HELP go_memstats_heap_released_bytes Number of heap bytes released to OS.
```

We verify the targets on prometheus dashboard are up and running as follows;

Prometheus Time Series

192.168.1.18:9090/targets

Prometheus Alerts Graph Status Help Classic UI

100/metrics		job="node-exporter"	ago		0:9100/metrics": dial tcp 192.168.1.20:9100: connect: no route to host

prometheus (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://192.168.1.18:9090/metrics	UP	instance="192.168.1.18:9090" job="prometheus"	382.000 ms ago	11.809ms	

windows-exporter (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://192.168.1.12:9182/metrics	UP	instance="192.168.1.12:9182" job="windows-exporter"	4.621s ago	953.391ms	