

Tugas Peserta:

1. Siapkan 2 server.
2. Server 2: install *Node Exporter*.
3. Server 1: install *Prometheus*.
4. Hubungkan *Node Exporter* pada Server 2 dengan *Prometheus* di Server 1.
5. Ambil dan tampilkan data *memory usage* dari Server 2 melalui *Prometheus*.

1. Buat 2 instance

<input type="checkbox"/>	Name 	Instance ID	Instance state	Instance type
<input type="checkbox"/>	valin-prometheus	i-0bee1a59ccc3ddb71	 Running  	t3.micro
<input type="checkbox"/>	valin-nodeexporter	i-0a2659fefd4786161	 Running  	t3.micro

2. SSH ke server ke 2 (valin-nodeexporter)

```
PS C:\Users\Valin\Downloads> ssh -i valin.pem ubuntu@13.212.165.187
The authenticity of host '13.212.165.187 (13.212.165.187)' can't be established.
ED25519 key fingerprint is SHA256:Ew/dy1ZtRhpeJRcU00IXn6KKl1HQyrfkQLJrlKLwmQc.
This key is not known by any other names.
```

3. Ikuti Langkah 1-12 di link ini (setting prometheus)

<https://medium.com/@shyamsandeep28/installation-of-prometheus-grafana-with-node-exporter-8ce3c97d968c>

Jangan lupa untuk add inbound rule di security group

Edit inbound rules

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules 

Inbound rule 1

Delete

Security group rule ID sgr-015c09465ad6fb58	Type  SSH	Protocol  TCP
Port range  22	Source type  Custom	Source  <input type="text" value="0.0.0.0"/> 

The screenshot shows the Prometheus Targets page. At the top, there are navigation icons and a status bar indicating 'Not secure' and the URL '13.251.81.219:9090/targets'. Below the header, there are links for Prometheus, Alerts, Graph, Status, Help, and Classic UI. The main section is titled 'Targets' and has three buttons: 'All' (selected), 'Unhealthy', and 'Collapse All'. A summary row for 'prometheus (1/1 up)' is shown, with a 'show less' link. The table below lists one endpoint: 'http://localhost:9090/metrics' which is 'UP' and has labels 'instance="localhost:9090" job="prometheus"'. The last scrape was 2.283s ago and it took 4.845ms.

Targets

All Unhealthy Collapse All

prometheus (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://localhost:9090/metrics	UP	instance="localhost:9090" job="prometheus"	2.283s ago	4.845ms	

4. Install node exporter di server ke 2 – ikuti dari langkah 15- di link ini :

<https://medium.com/@shyamsandeep28/installation-of-prometheus-grafana-with-node-exporter-8ce3c97d968c>

Jangan lupa open port 9100 di security group

The screenshot shows the AWS CloudFormation Inbound rule configuration for a security group. The rule is named 'Inbound rule 2'. It has the following details:

- Security group rule ID:** sgr-05e84002f6817ea17
- Type:** Custom TCP
- Protocol:** TCP
- Port range:** 9100
- Source type:** Custom
- Source:** 0.0.0.0/0

5. Periksa bahwa node exporter berjalan dengan membuka:

The screenshot shows the output of a curl command to a Node Exporter instance at '13.212.165.187:9100/metrics'. The output is a list of metrics in plain text, including:

```
# 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 0
# HELP go_goroutines Number of goroutines that currently exist.
# TYPE go_goroutines gauge
go_goroutines 9
# HELP go_info Information about the Go environment.
# TYPE go_info gauge
go_info{version="go1.16.6"} 1
# HELP go_memstats_alloc_bytes Number of bytes allocated and still in use.
# TYPE go_memstats_alloc_bytes gauge
go_memstats_alloc_bytes 1.33024e+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 1.33024e+06
# 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
```

6. Configure the Node Exporter as a Prometheus target

Ikuti step mulai dari nomor 17 di link ini :

<https://medium.com/@shyamsandeep28/installation-of-prometheus-grafana-with-node-exporter-8ce3c97d968c>

Di prometheus.yml, ubah di bagian static configsnya

```
ubuntu@ip-172-31-1-29: /etc/ ~ + ^

GNU nano 7.2                                     prometheus.yml
# scrape_timeout is set to the global default (10s).

# Alertmanager configuration
alerting:
  alertmanagers:
    - static_configs:
      - targets:
          # - alertmanager:9093

# Load rules once and periodically evaluate them according to the global 'evaluation_interval'.
rule_files:
  # - "first_rules.yml"
  # - "second_rules.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 from this config.
  - job_name: 'prometheus'

    # metrics_path defaults to '/metrics'
    # scheme defaults to 'http'.

    static_configs:
      - targets: ['localhost:9090', 'localhost:9100', '13.212.165.187:9100']
```

The screenshot shows the Prometheus UI with the URL `13.251.81.219:9090/targets`. The page title is "Targets". There are three targets listed:

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
<code>http://localhost:9090/metrics</code>	UP	<code>instance="localhost:9090" job="prometheus"</code>	12.226s ago	6.043ms	
<code>http://localhost:9100/metrics</code>	DOWN	<code>instance="localhost:9100" job="prometheus"</code>	4.753s ago	0.379ms	Get "http://localhost:9100/metrics": dial tcp 127.0.0.1:9100: connect: connection refused
<code>http://13.212.165.187:9100/metrics</code>	UP	<code>instance="13.212.165.187:9100" job="prometheus"</code>	1.292s ago	14.715ms	

Alasan localhost:9100 DOWN – karena memang tidak ada node exporter di server prometheus, bagian localhost:9100 di static config bisa dihapus saja.

7. Tampilkan data memory usage dari server 2 di prometheus

