

Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Vacaciones diciembre 2020
Laboratorio de Sistemas Operativos 2

Robinson Jonathan Pérez Fuentes 201114056
Luis Ricardo Hernández 201114490

Manual de Monitoreo

Para el monitoreo del sistema se configurará la herramienta de Grafana y a través de Prometheus se extraerán los datos para que se alimente.

Requerimientos De sistema

Para configurar esta herramienta se instala sobre una máquina virtual en la nube con los siguientes requerimientos:

- Sistema operativo Ubuntu 20.04 LTS
- 2 CPUs tipo E2 medium
- 4 GB memoria RAM
- 25 GB de almacenamiento

Proceso de instalación

Para iniciar la instalación actualizar el equipo con los siguientes comandos:

```
sudo apt update  
sudo apt upgrade
```

Instalar un sistema para edición de archivos

```
sudo apt install nano
```

Descargar el software, para descargar desde consola con el siguiente comando, si se desea instalar otra versión se puede verificar en la página oficial

<https://prometheus.io/download/>

```
wget
```

<https://github.com/prometheus/prometheus/releases/download/v2.23.0/prometheus-2.23.0.linux-amd64.tar.gz>

Para descomprimir el archive en el equipo:

```
tar xzf prometheus-2.23.0.linux-amd64.tar.gz
```

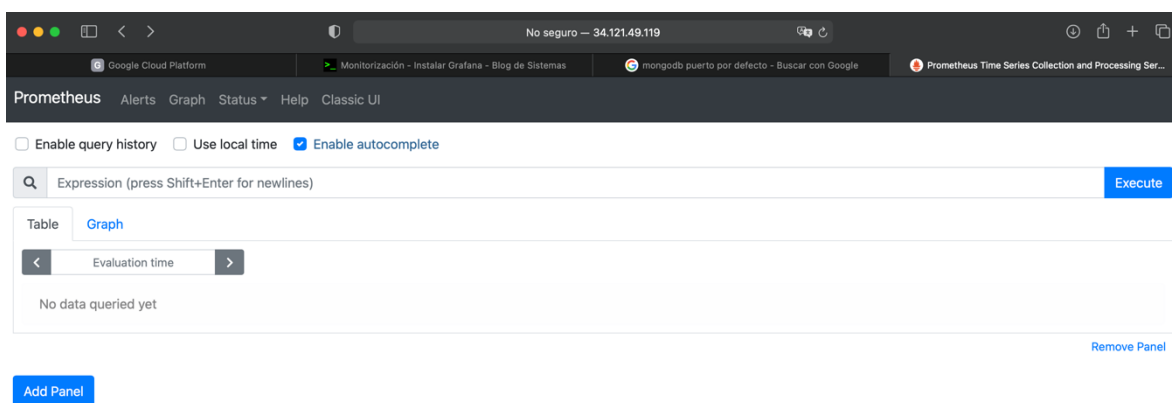
Moverse al directorio

```
cd prometheus-2.23.0.linux-amd64.tar.gz
```

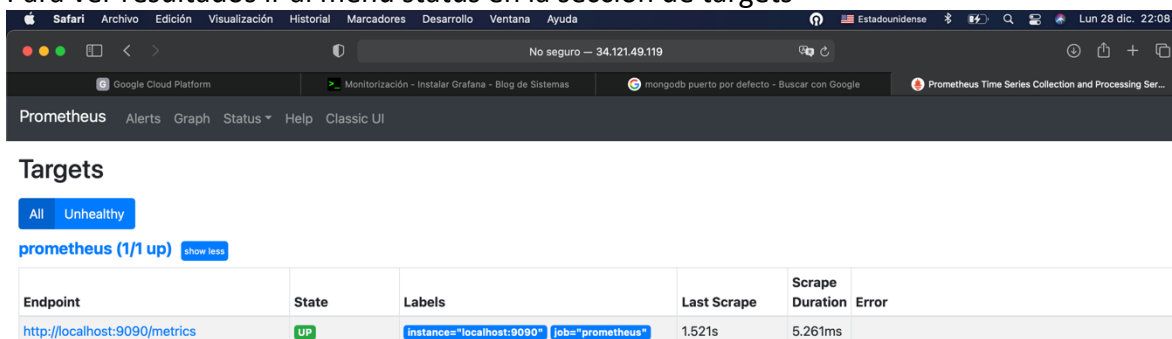
Iniciar prometheus con el siguiente comando.

```
./prometheus
```

<http://34.121.49.119:9090>



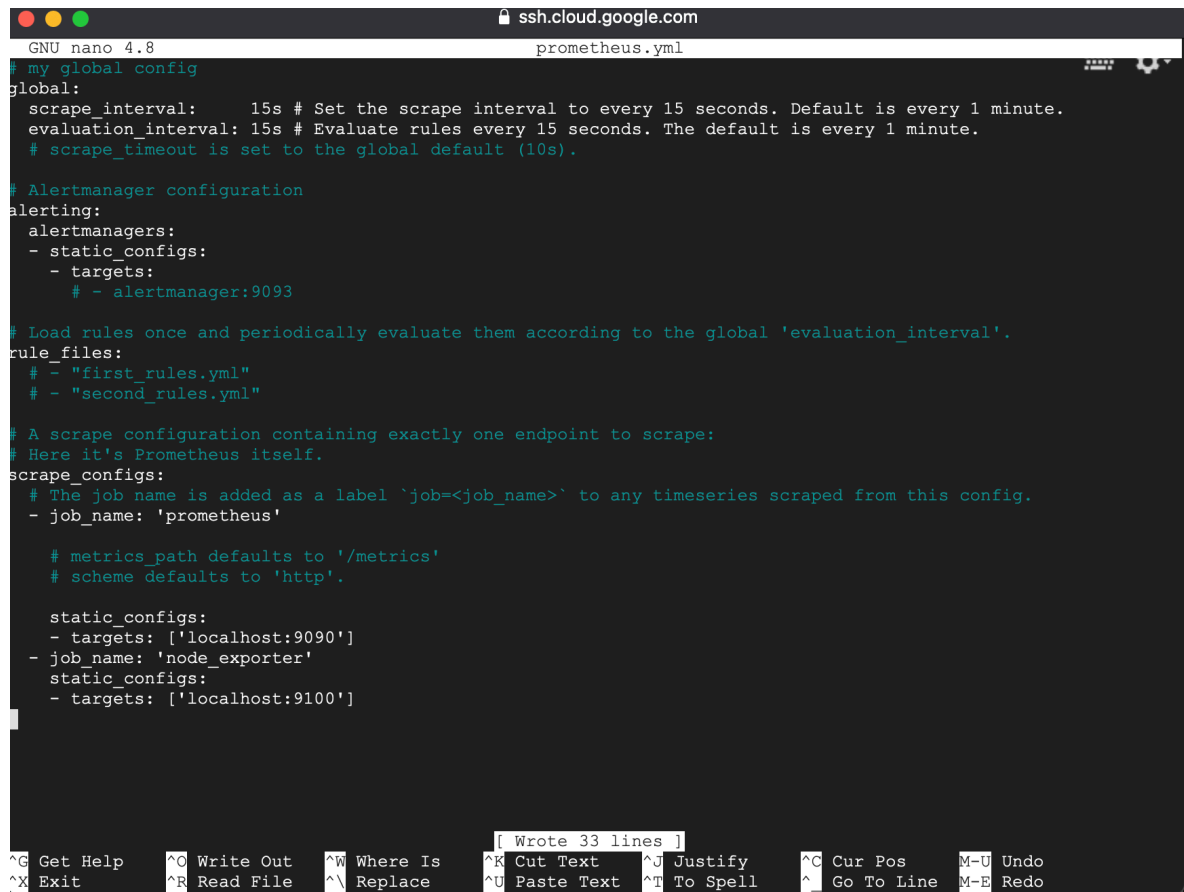
Para ver resultados ir al menú status en la sección de targets



```
# 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 11
# HELP go_info Information about the Go environment.
# TYPE go_info gauge
go_info(version="go1.14.4") 1
# HELP go_memstats_alloc_bytes Number of bytes allocated and still in use.
# TYPE go_memstats_alloc_bytes gauge
go_memstats_alloc_bytes 1.241936e+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 1.241936e+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
go_memstats_buck_hash_sys_bytes 1.444815e+06
# HELP go_memstats_frees_total Total number of frees.
# TYPE go_memstats_frees_total counter
go_memstats_frees_total 634
# 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
# 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 3.436808e+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 1.241936e+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 6.444646e+07
# 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 2.33472e+06
# HELP go_memstats_heap_objects Number of allocated objects.
# TYPE go_memstats_heap_objects gauge
go_memstats_heap_objects 7526
# HELP go_memstats_heap_released_bytes Number of heap bytes released to OS.
# TYPE go_memstats_heap_released_bytes gauge
go_memstats_heap_released_bytes 6.444646e+07
# HELP go_memstats_heap_sys_bytes Number of heap bytes obtained from system.
# TYPE go_memstats_heap_sys_bytes gauge
go_memstats_heap_sys_bytes 6.478184e+07
# HELP go_memstats_last_gc_time_seconds Number of seconds since 1970 of last garbage collection.
```

Luego de eso modificar el yml de prometheus

nano prometheus.yml



```
GNU nano 4.8 prometheus.yml
# my global config
global:
  scrape_interval: 15s # Set the scrape interval to every 15 seconds. Default is every 1 minute.
  evaluation_interval: 15s # Evaluate rules every 15 seconds. The default is every 1 minute.
  # 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']
  - job_name: 'node_exporter'
    static_configs:
      - targets: ['localhost:9100']
```

Ejecutar de nuevo prometheus:

./prometheus

Instalar grafana

Para instalar ir a la pagina oficial y obtener el enlace para el sistema operativo correr el siguiente comando:

wget <https://dl.grafana.com/oss/release/grafana-7.3.6.linux-amd64.tar.gz>

Descomprimir el archive:

tar -zxvf grafana-7.3.6.linux-amd64.tar.gz

Moverse a la carpeta

```
cd grafana-7.3.6/  
cd bin/
```

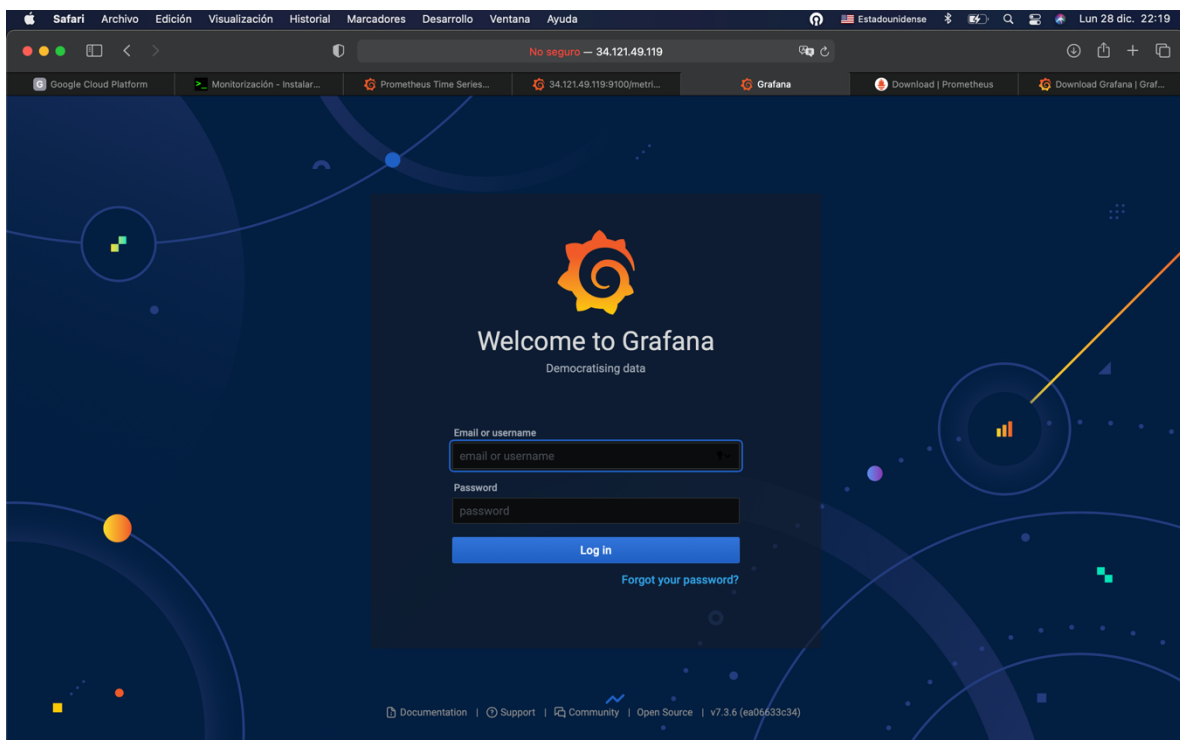
Ejecutar grafana

```
./grafana-server
```

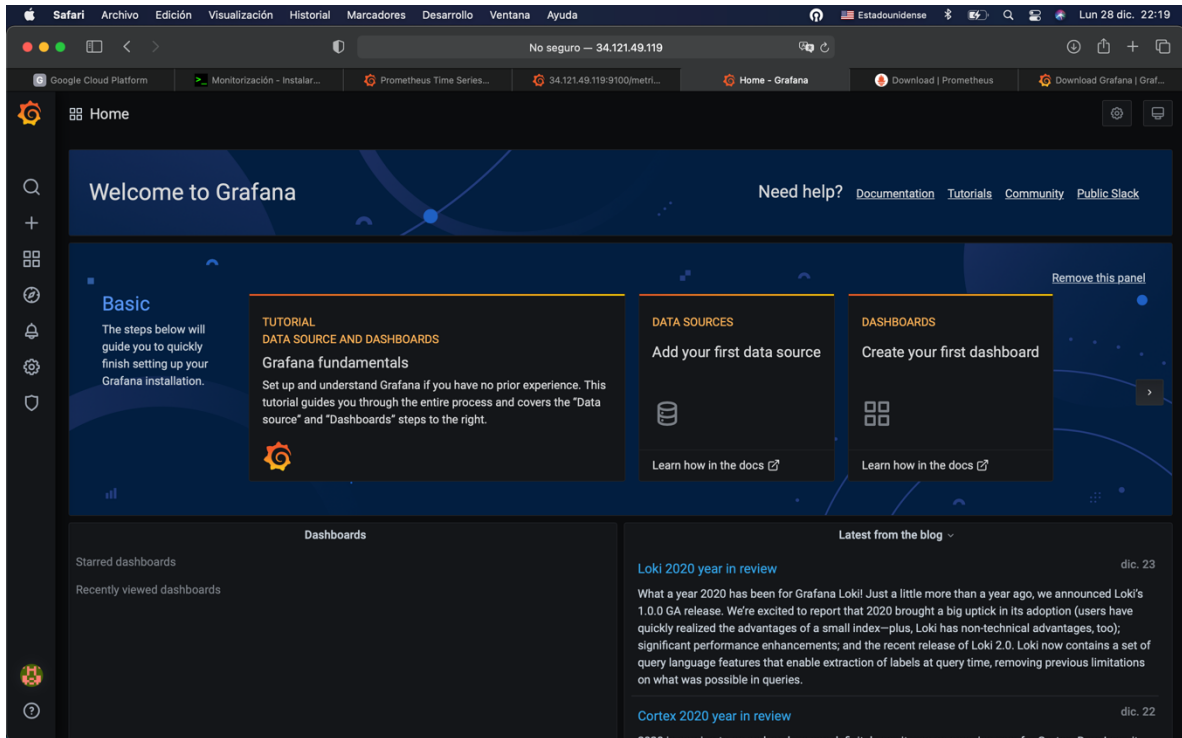
#para habilitar un dashboard

<https://grafana.com/grafana/dashboards/1860>

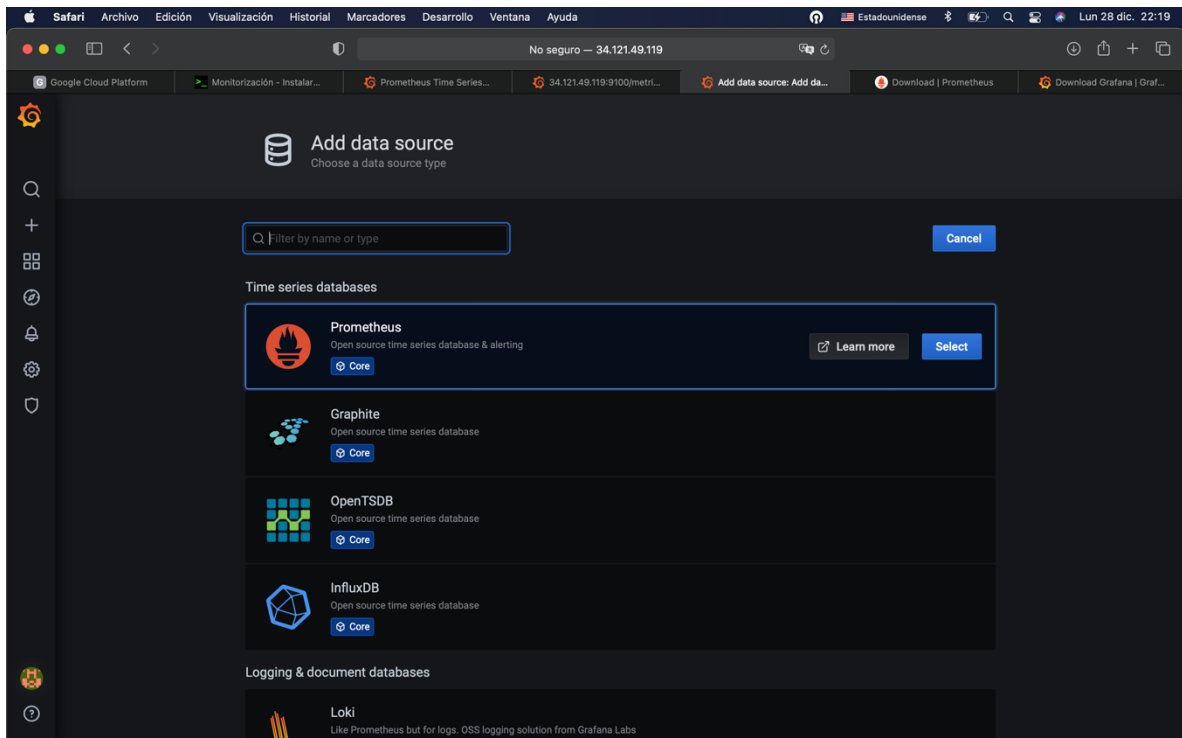
El usuario y password por defecto es admin admin



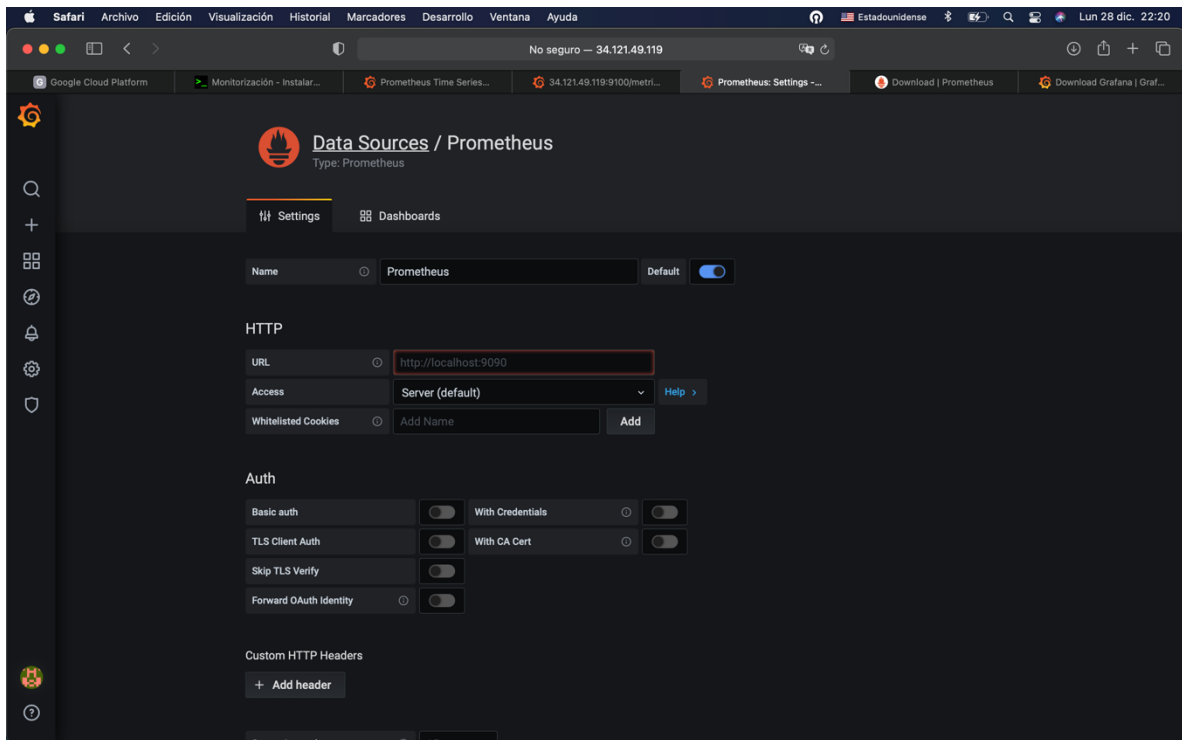
Hacer login



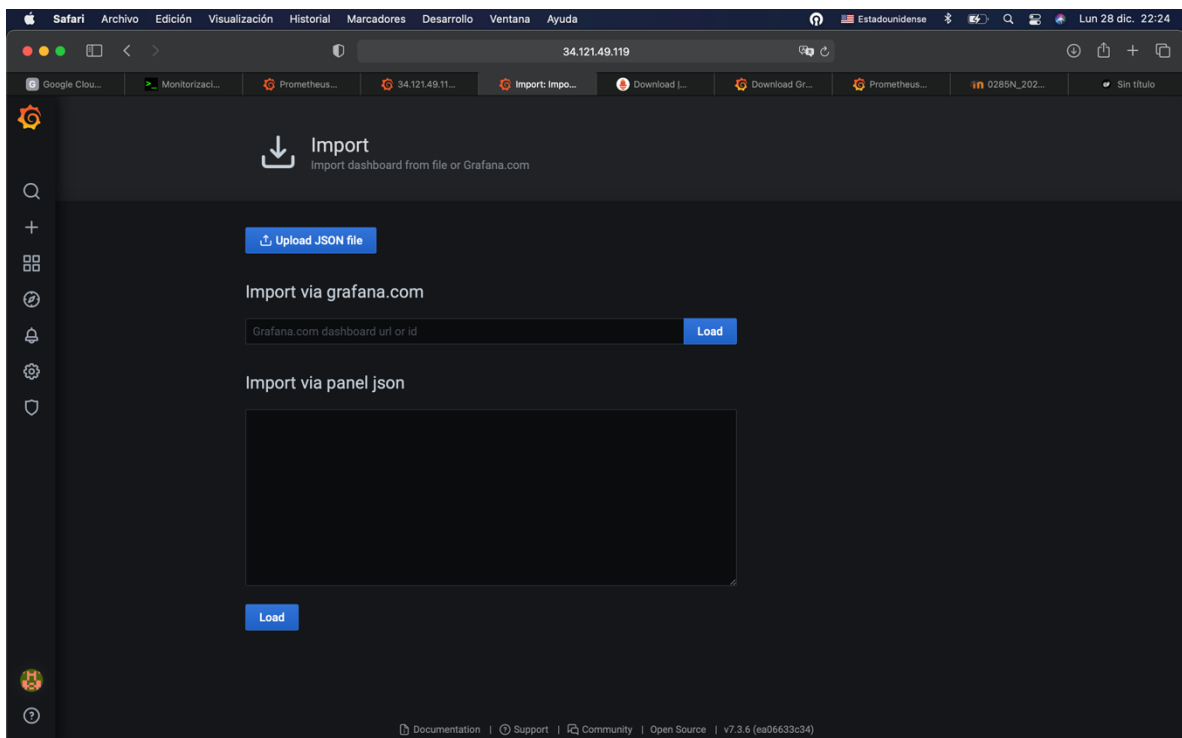
Agregar un source



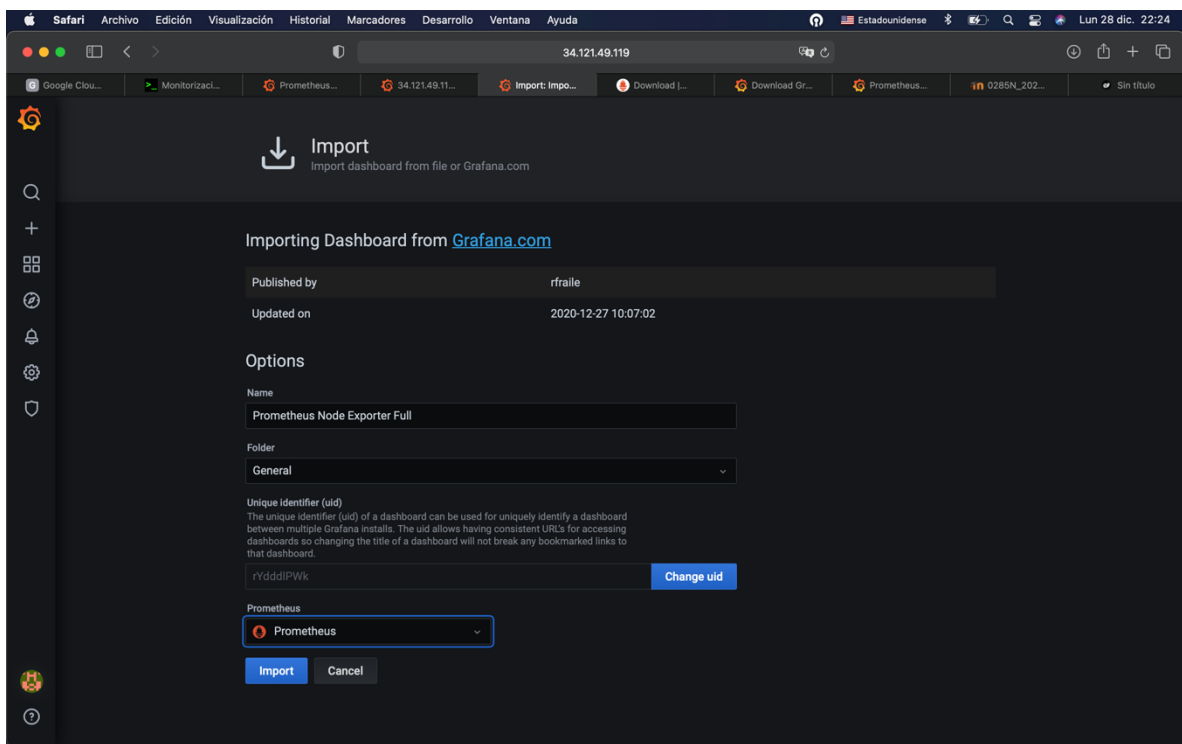
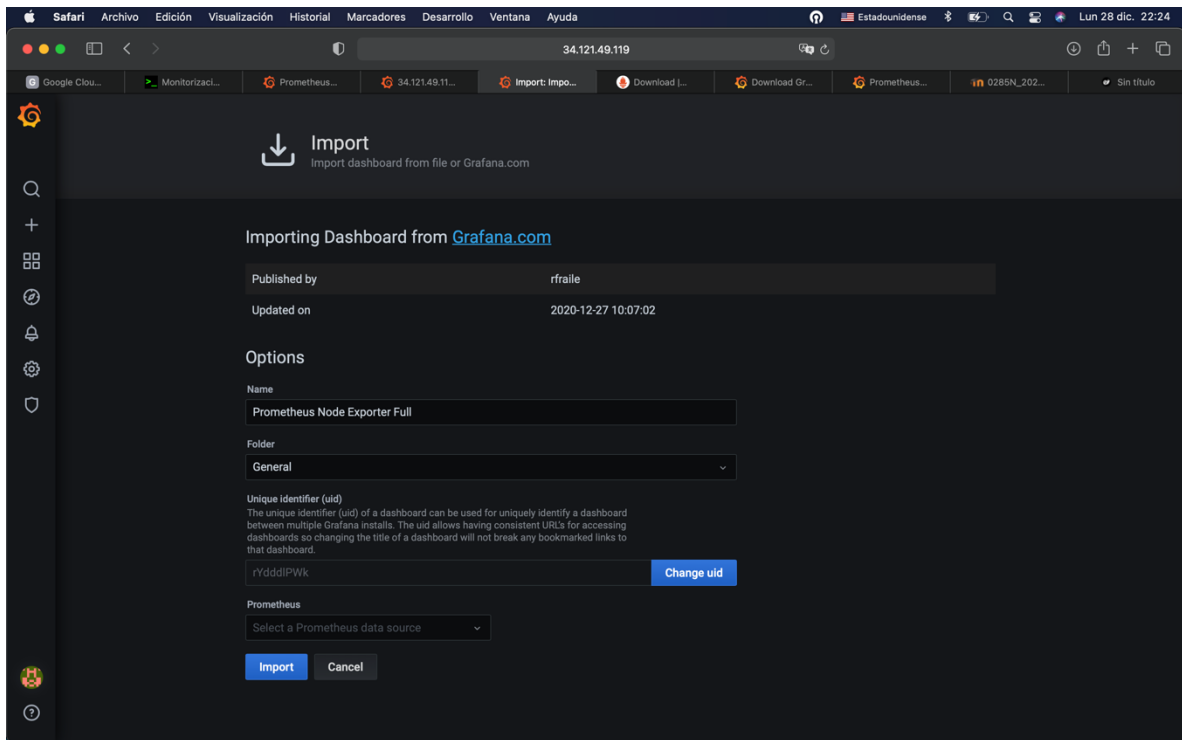
Seleccionar prometheus:



Agregar el id i cargar



Importar y seleccionar de nuevo prometheus



En datasource dejar default, en job node_exporter y en host la ip:9100

