



Noviembre de 2022
Laboratorio 6
Monitoreo y supervisión de redes

INTRODUCCIÓN

Topología

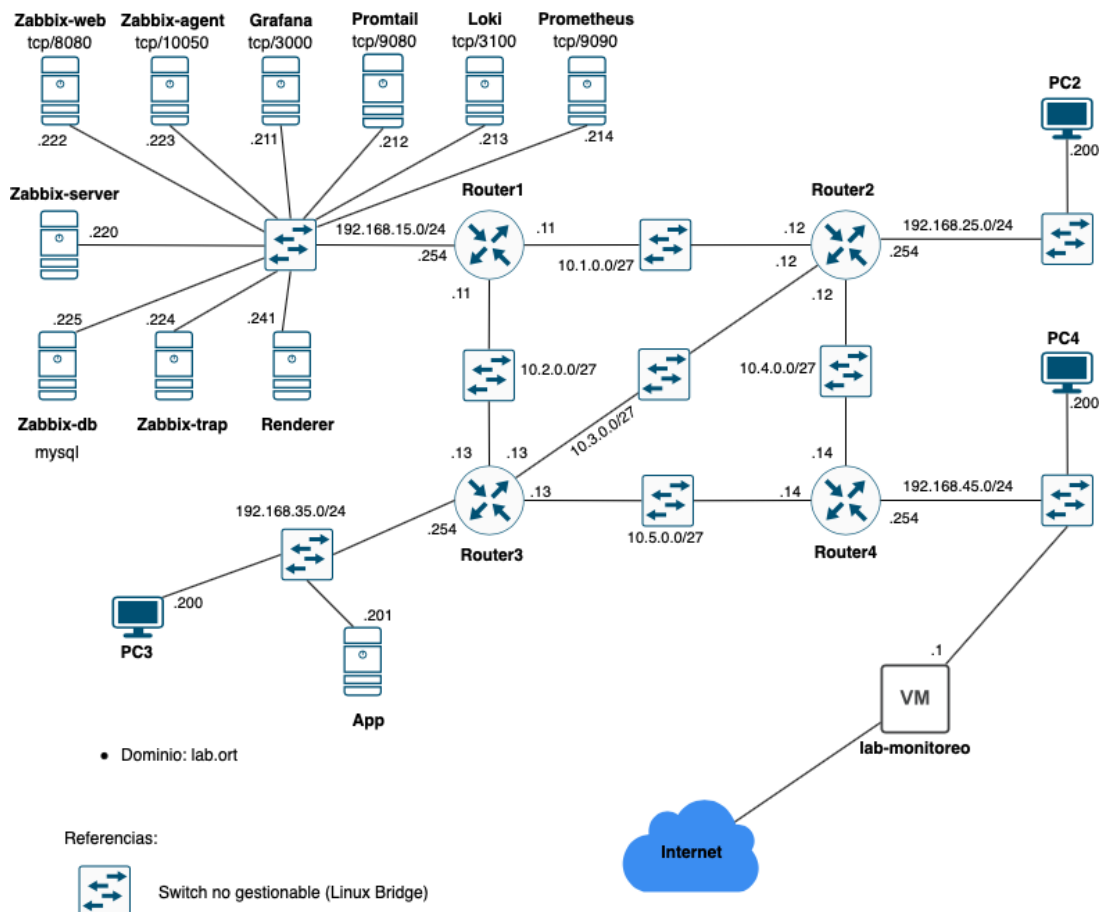


Imagen 1: Topología de red

La conexión se realiza mediante SSH a la máquina virtual lab-monitoreo. Por defecto se presenta en el puerto tcp/2222 de la interface de red de la computadora personal del estudiante

Usuario	estudiante
Password	estudiante
Puerto	2222

1. Conectarse por SSH a la VM lab-monitoreo con redirección X11 habilitada
2. Si ingresa con MobaXterm viene habilitado por defecto

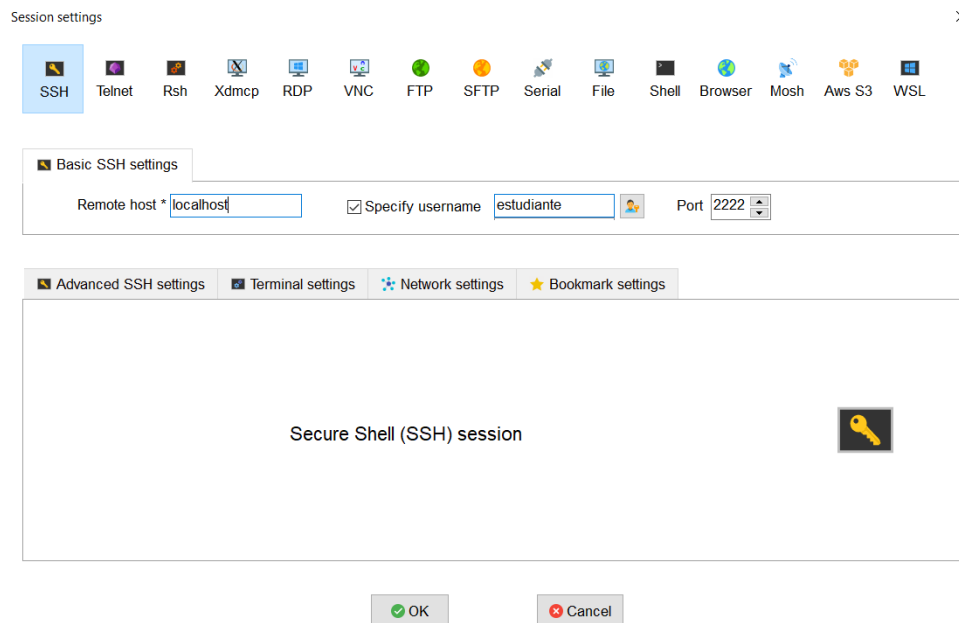


Imagen 2: Ejemplo en MobaXterm

3. Para habilitar X11 en Putty:

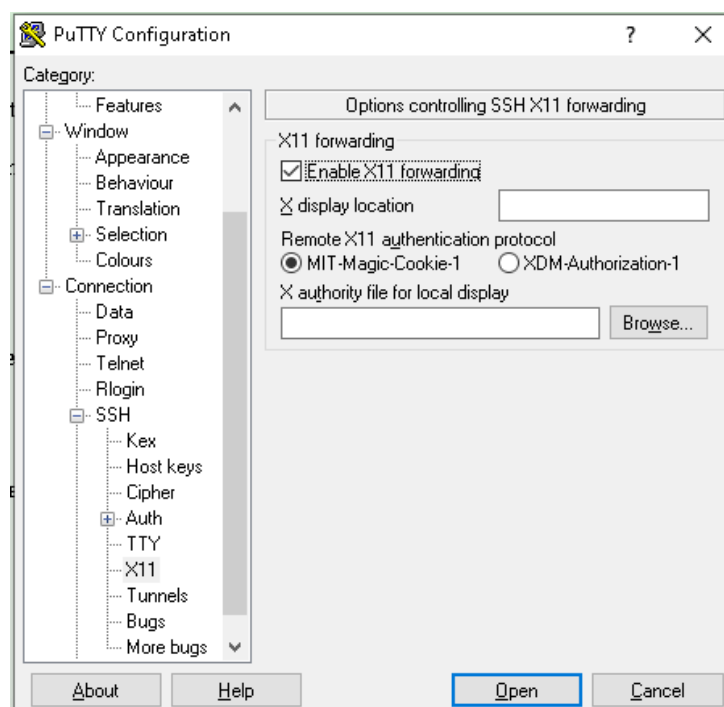


Imagen 3: Configuración Putty

4. Primero que nada prepararemos el ambiente para este laboratorio, ejecutemos:

- `lab6`

5. En el navegador de nuestra PC ingresemos a <http://localhost:2225>

Se trata de una aplicación de prueba que permite postular y votar sitios web, en la cual nos basaremos para realizar pruebas de monitoreo

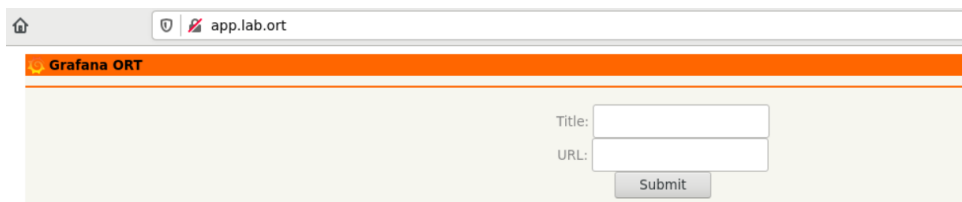


Imagen 4: App de prueba

6. En otra pestaña ingresemos a Grafana: <http://localhost:2224>

Credenciales de acceso:

Usuario	admin
Password	admin

PRÁCTICOS

Práctico 1

En este práctico veremos cómo consumir datos de un servidor Prometheus desde Grafana

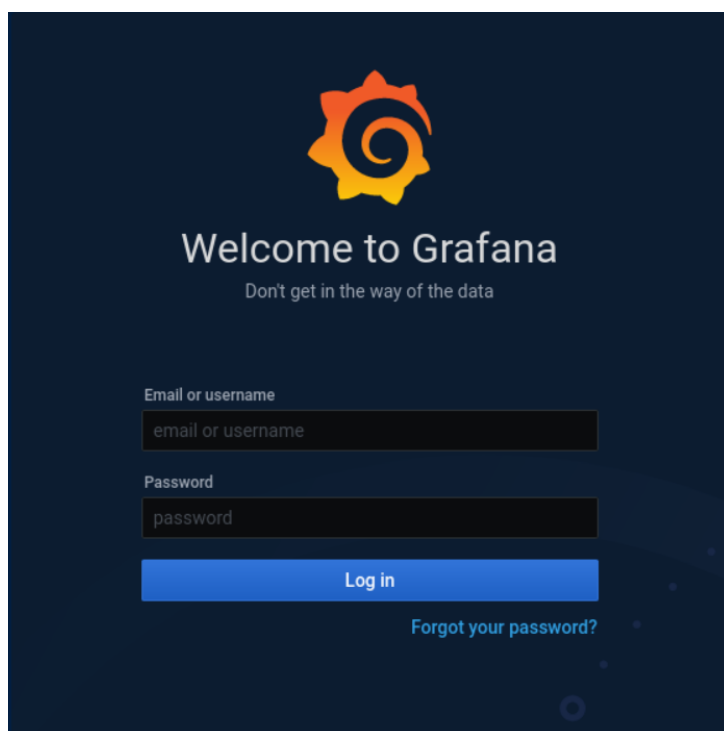


Imagen 5: Login Grafana

1. Lo primero que haremos es agregar un Data Source:

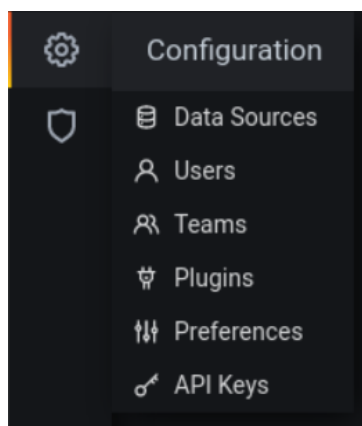
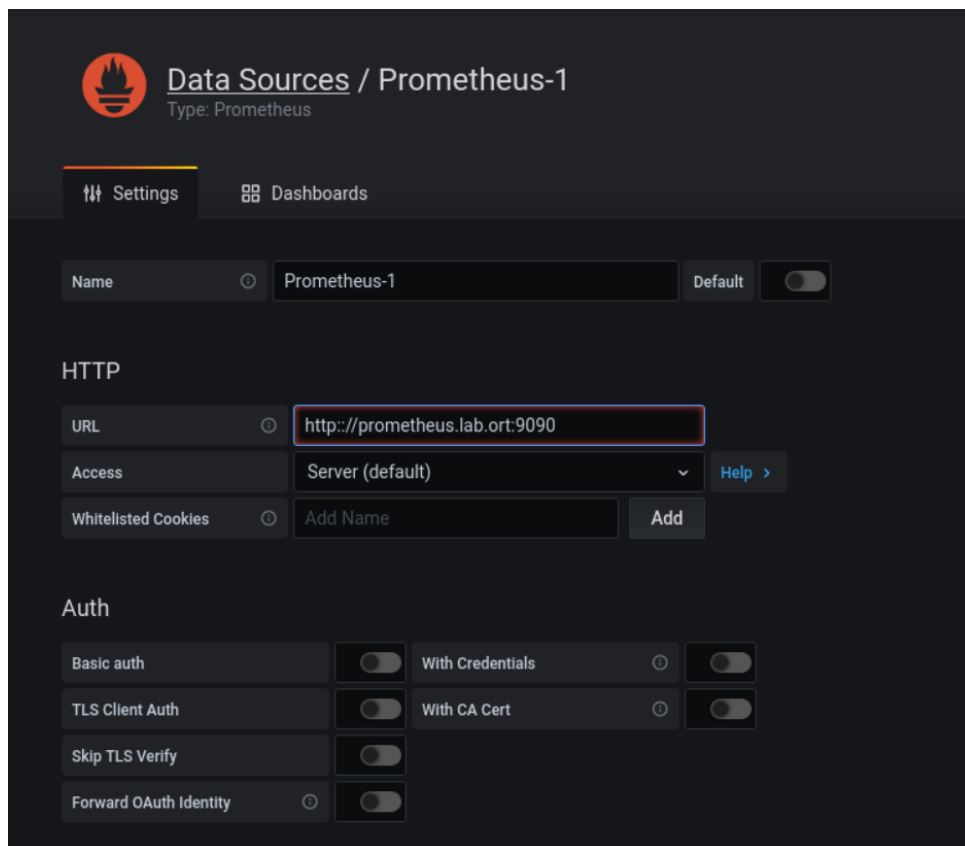


Imagen 6: Grafana

2. Seleccionamos Prometheus



Data Sources / Prometheus-1
Type: Prometheus

Settings | Dashboards

Name: Prometheus-1 Default ☐

HTTP

URL: http://prometheus.lab.ort:9090

Access: Server (default) Help >

Whitelisted Cookies: Add Name Add

Auth

Basic auth ☐ With Credentials ☐

TLS Client Auth ☐ With CA Cert ☐

Skip TLS Verify ☐

Forward OAuth Identity ☐

Imagen 7: Grafana

- URL: `http://prometheus.lab.ort:9090`

3. Clickeamos en Save & Test, si los datos están correctos debería aparecer el mensaje "Data source is working"

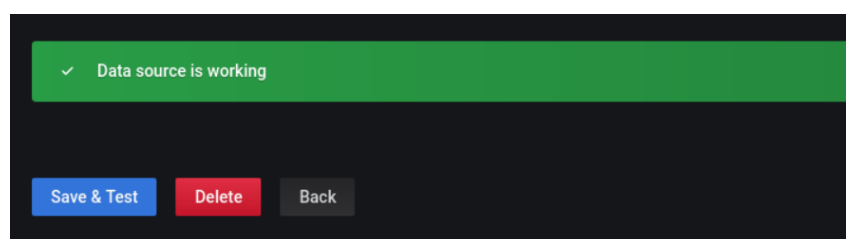


Imagen 8: Grafana

4. Exploremos los datos obtenidos por Prometheus, para eso vayamos a Explore

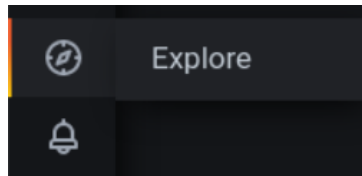


Imagen 9: Grafana

- Usaremos el lenguaje PromQL
(<https://prometheus.io/docs/prometheus/latest/querying/basics/>)

5. Probemos realizar las siguientes consultas, utilizando este panel

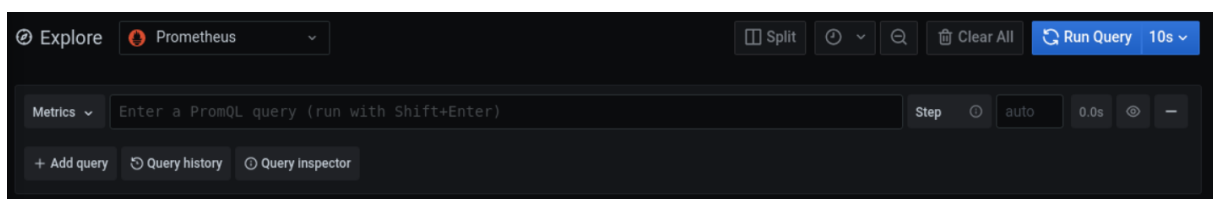


Imagen 10: Grafana

- (a) `tns_request_duration_seconds_count`: es un contador que siempre incrementa
- (b) `rate(tns_request_duration_seconds_count[5m])`: la tasa de cambio del contador
- (c) `sum(rate(tns_request_duration_seconds_count[5m])) by (route)`: la tasa de cambio agrupada por el parámetro `route`
- (d) Interactúe con la aplicación: definiendo nuevas URL, recargando la página o votando por algún link para observar la variación de las gráficas
- (e) Pruebe a cambiar el rango de tiempo, para tener mayor o menor definición
- (f) Pruebe a definir un intervalo de actualización automática

Práctico 2

Grafana/Loki

En este práctico veremos cómo agregar a Grafana datos de un servidor Loki

1. Agreguemos un nuevo Data Source:

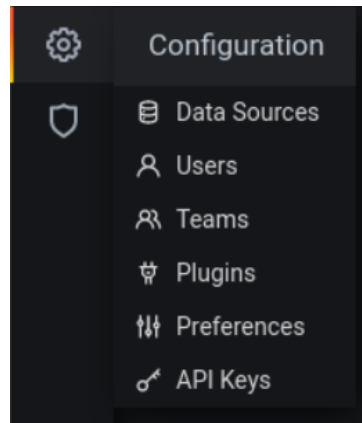


Imagen 11: Grafana

2. Seleccionamos Loki

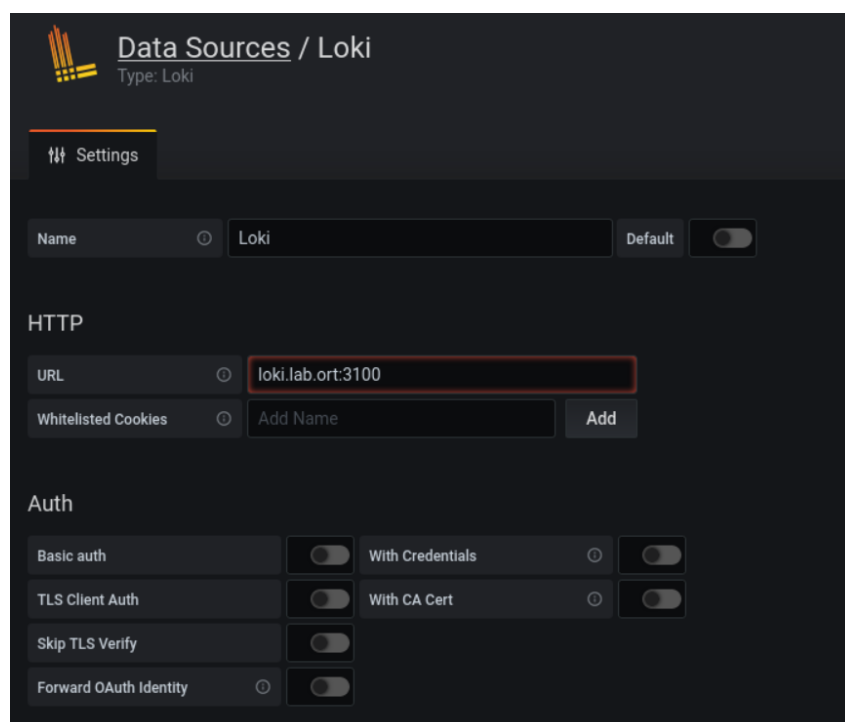


Imagen 12: Grafana

- URL: http://loki.lab.ort:3100

3. Clickeamos en Save & Test, si los datos están correctos debería aparecer el mensaje "Data source is working"

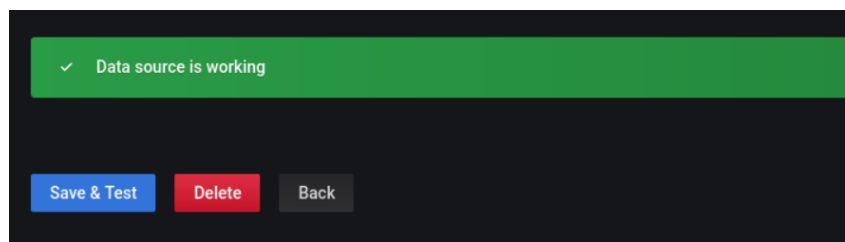


Imagen 13: Grafana

4. Exploremos los datos obtenidos por Loki, para eso vayamos a Explore

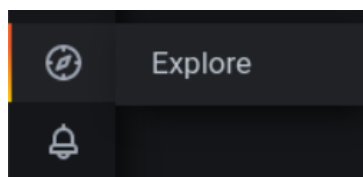


Imagen 14: Grafana

5. Para realizar consultas usaremos el lenguaje LogQL (<https://grafana.com/docs/loki/latest/logql/>)

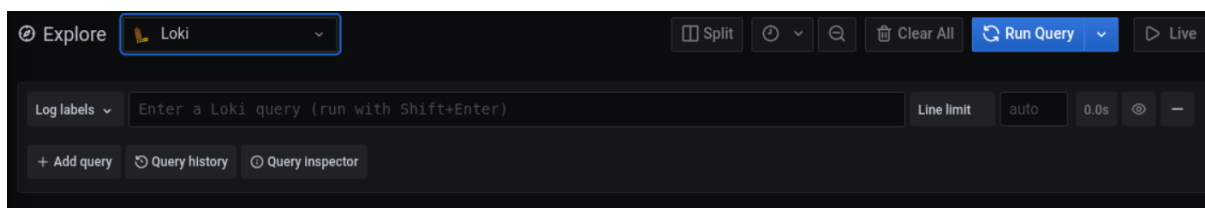


Imagen 15: Grafana

6. Probemos a realizar la siguiente consultas:
 - (a) `{filename="/var/log/tns-app.log"}`: Nos mostrará todo el contenido del archivo `/var/log/tns-app.log` en el período seleccionado

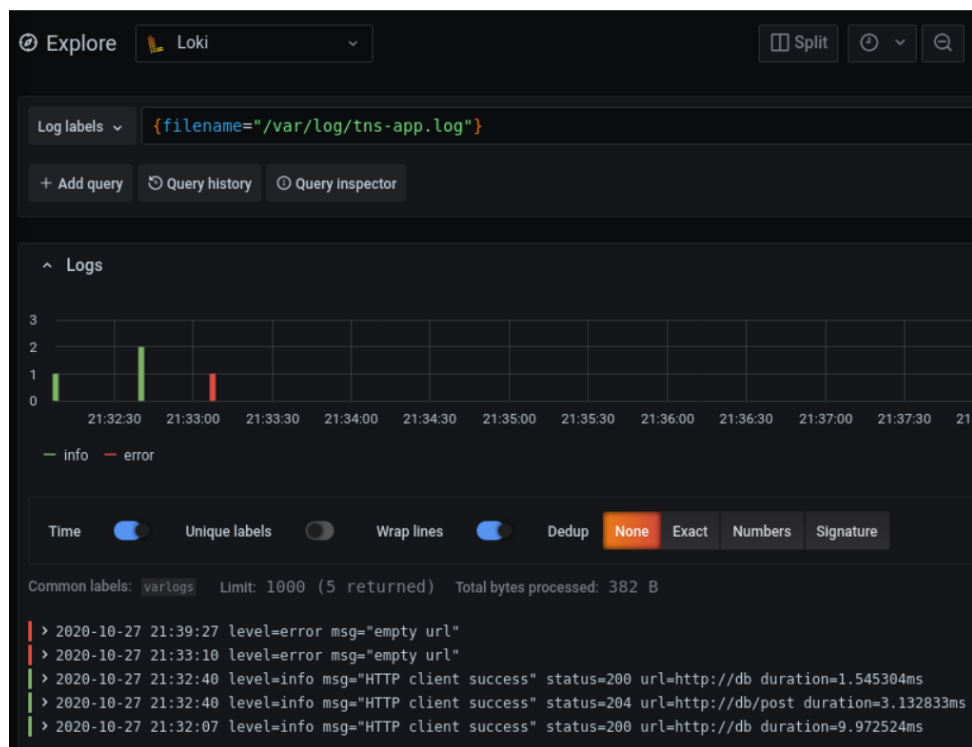


Imagen 16: Grafana

- (b) {filename="/var/log/tns-app.log"} |= "error": Nos mostrará todas las líneas que contengan el string "error" dentro del archivo /var/log/tns-app.log
- Para generar un error ingrese en app.lab.ort una nueva web con el campo URL en blanco

Práctico 3

Grafana Dashboard

En este práctico veremos cómo crear dashboards en Grafana

1. Vayamos a crear un Dashboard

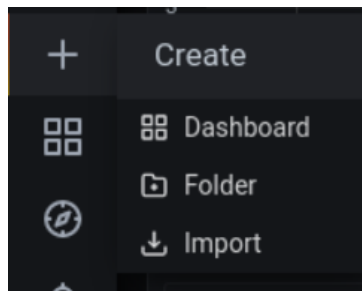


Imagen 17: Grafana

2. Agreguemos un nuevo panel seleccionando Add new panel

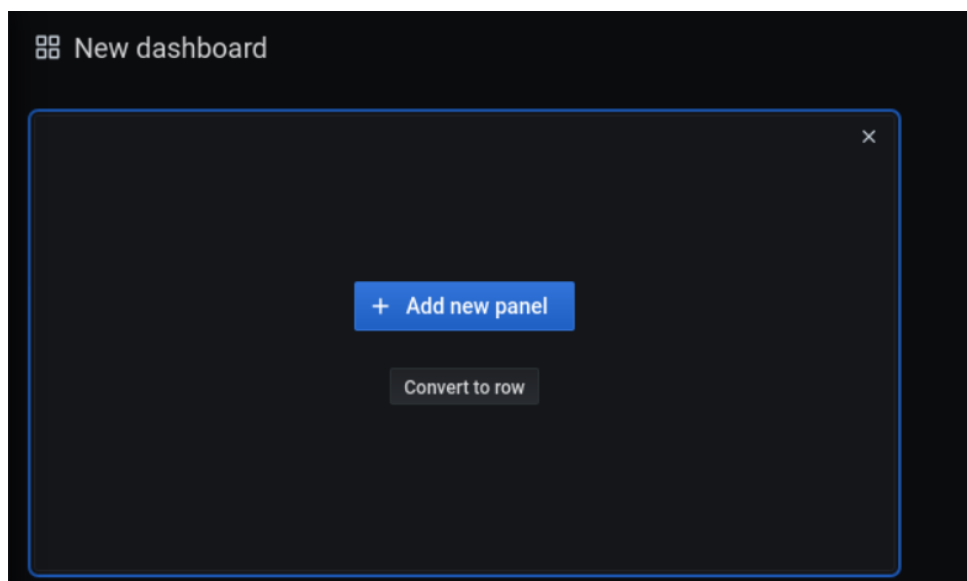


Imagen 18: Grafana

3. Definamos los criterios para realizar la gráfica

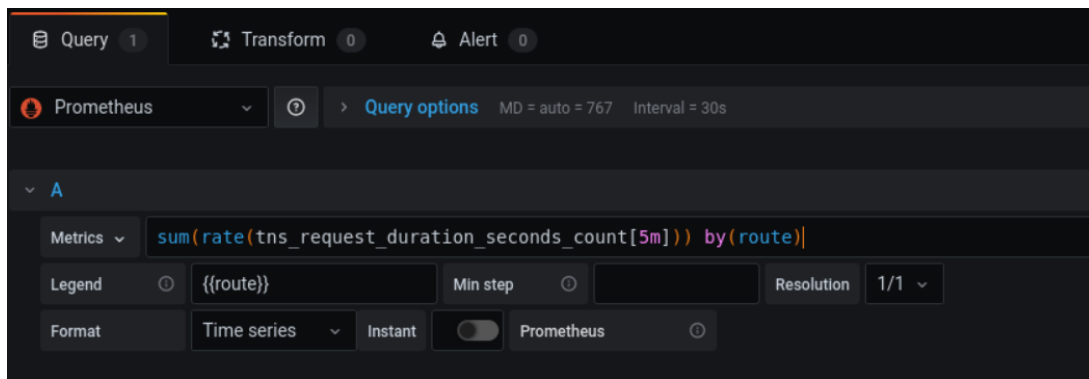


Imagen 19: Grafana

- Metrics: `sum(rate(tns_request_duration_seconds_count[5m])) by(route)`
- Legend: `{{route}}`

4. Hagamos click en Apply para guardar los cambios

Práctico 4

Grafana Annotations

El objetivo es entender cómo crear notas en una gráfica de Grafana

1. Para agregar una nota manual hagamos click en alguna parte de la gráfica y hagamos click en Add annotation
2. Probemos a crear una nota
3. Vayamos a la app y generemos algo de tráfico, creando nuevas web, recargando y/o votando
4. Revisemos la gráfica nuevamente, debería reflejarse el uso que le hayan dado a la app
5. Probemos ingresar una nota sobre la zona del evento (con la ayuda de la tecla Ctrl se puede seleccionar un rango)



Imagen 20: Grafana

6. Se pueden agregar notas de forma automática en base a una consulta determinada
7. Accedamos a la configuración del dashboard definido anteriormente (engranaje)

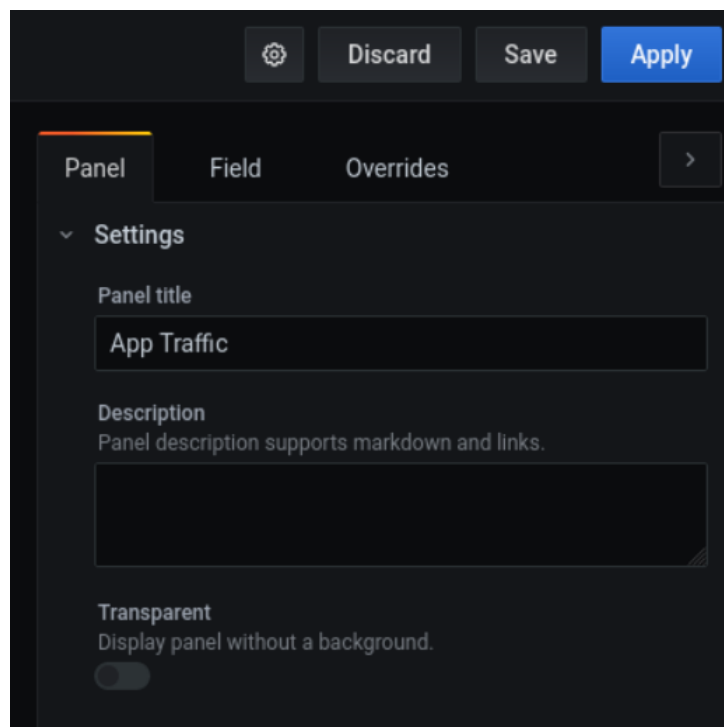


Imagen 21: Grafana

8. En Annotations configuremos notas en función de los eventos que registre Loki

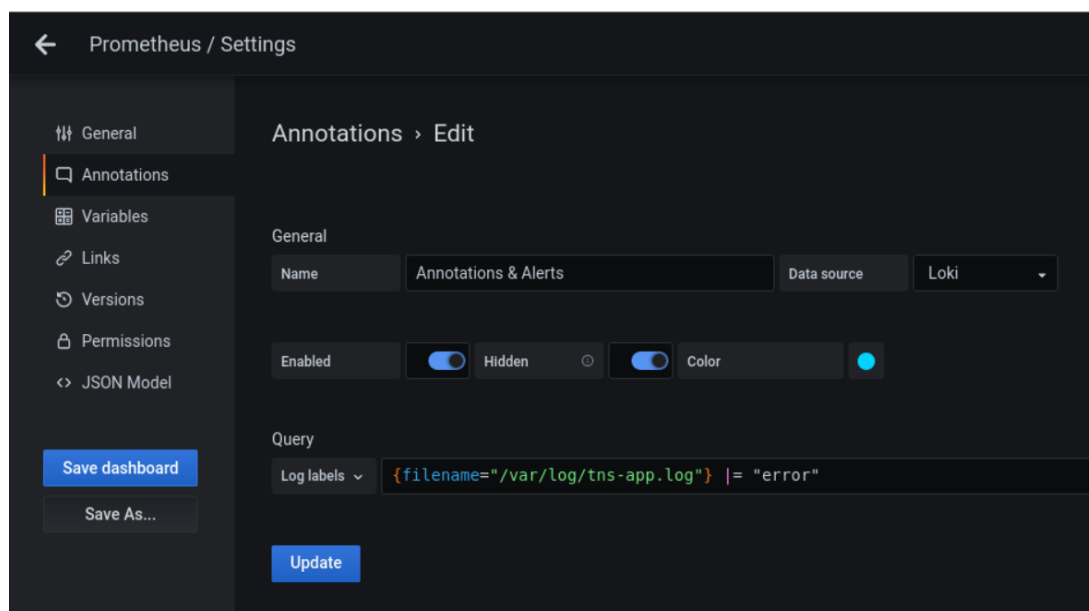


Imagen 22: Grafana

- Name: Errors
- Data Source: Loki
- Query: {filename="/var/log/tns-app.log"} |= "error"

9. Al volver al dashboard se deberían observar las anotaciones nuevas en cada momento que hubo un error.

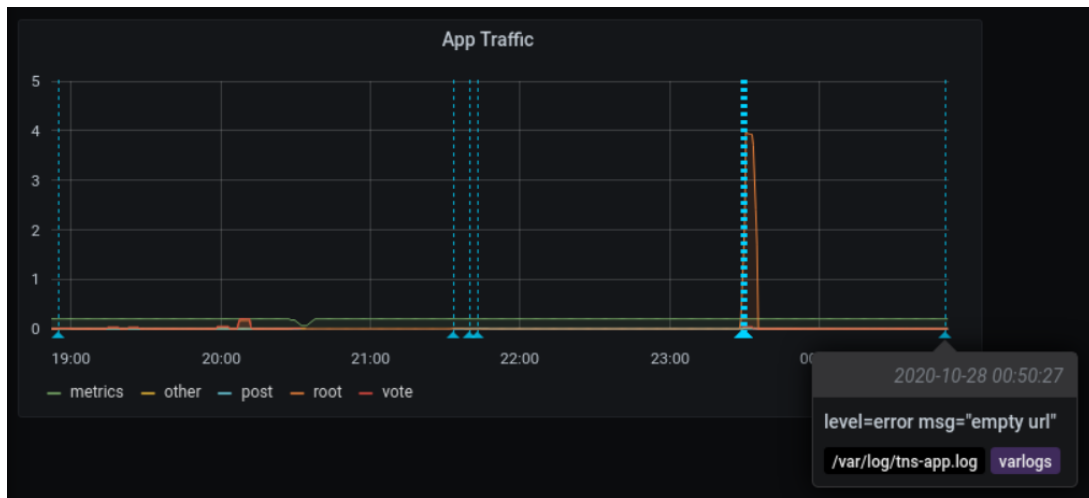


Imagen 23: Grafana