**Documentación del API de Farmacia Chávez**

**Autor:** Richard Poma  
**Fecha:** 18/10/2024

#### Índice

1. **Descripción del proyecto**
2. **Requisitos previos**
3. **Configuración del entorno**
4. **Implementación de API**
5. **Prueba y verificación de la API**
6. **Funcionamiento del sistema**
7. **Solución de Problemas Comunes**
8. Funcionalidades adicionales
9. Descripción del proyecto

Este proyecto está diseñado para monitorear en tiempo real los routers MikroTik de Farmacia Chávez. El sistema permite la recolección de métricas de rendimiento, la emisión de alertas en caso de fallos, la visualización de datos en Grafana y la automatización de reinicios en situaciones de error. El objetivo es asegurar el funcionamiento óptimo de la red mediante una gestión proactiva.

1. Requisitos previos

**Lenguaje y Frameworks:**

* Python (recomendado: versión 3.8 o superior)

Servicios y Librerías

* **Prometheus**: Para recolectar métricas de los routers MikroTik.
* **Grafana**: Para la visualización de métricas
* **Alertmanager**: Para gestionar alertas en tiempo real
* **Docker**: Para contenerizar los servicios y facilitar el despliegue.

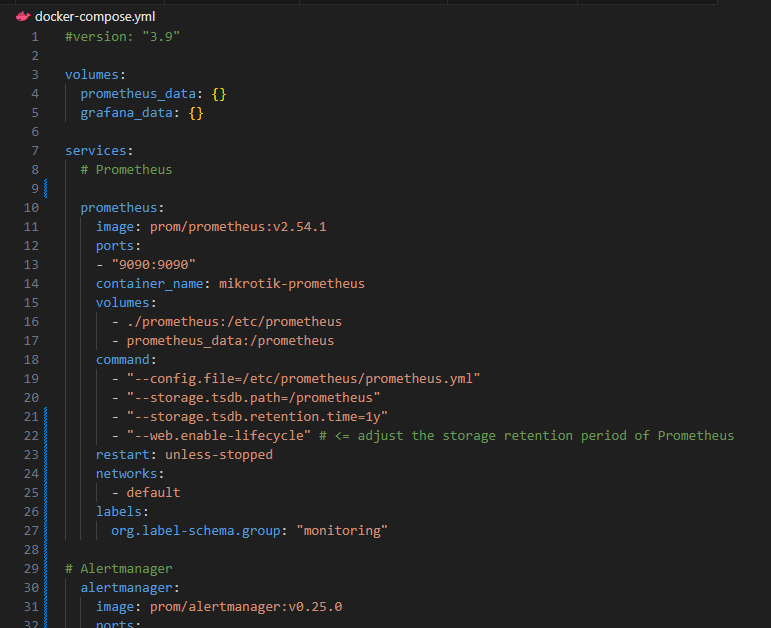
Herramientas adicionales

* **Docker Compose**: Para la orquestación de contenedores.
* **Activar WSL 2**: Docker Desktop de WSL 2 para ejecutar contenedores de Linux en Windows.
* **Window 10, 11**
* **API de Telegram**: Para la integración y envío de alertas a un grupo o usuario en Telegram

1. Configuración del entorno

Paso 1: Configuración de Docker Compose

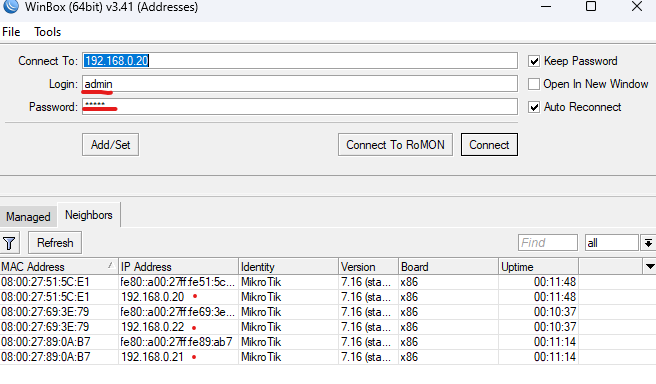
En el archivo “docker-compose.yml”, se configuro los contenedores necesarios para Prometheus, Grafana y Alertmanager. Aquí tienes un ejemplo de configuración básica:



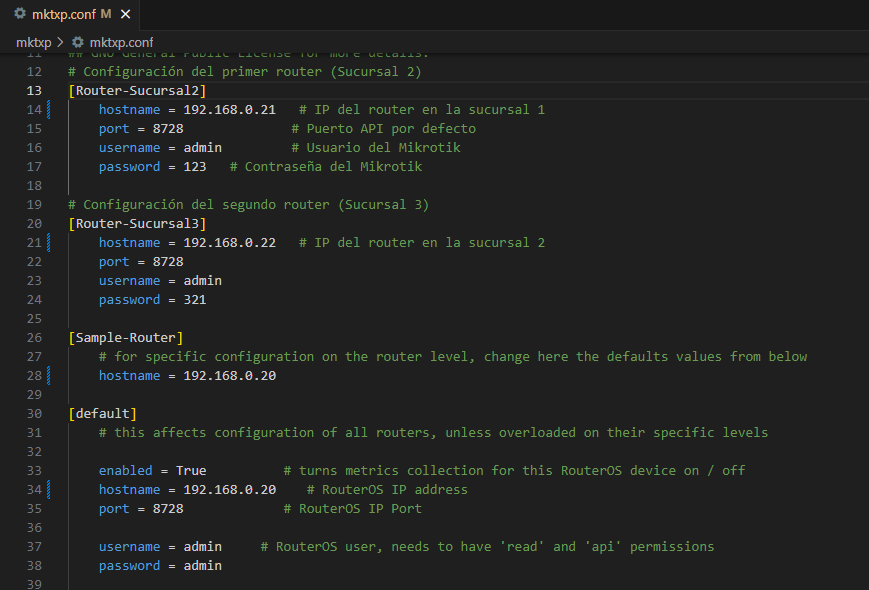
Paso 2: **Adición de Routers MikroTik**

Puede agregar cualquier cantidad de routers MikroTik que desee para el monitoreo. En este ejemplo, se han configurado tres routers de prueba, cada uno con su respectiva dirección IP y credenciales. Es fundamental que las IP y contraseñas en el archivo de configuración “mktxp.conf” sean correctas para garantizar una conexión estable y pruebas exitosas.

Comprobamos que en Winbox tenemos todos los Reuters de MikroTik



Reuters de MikroTiks IP y contraceñas.



Paso 3: Configuración de Telegram para Alertmanager

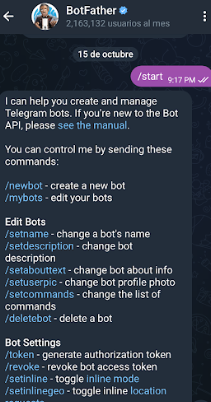
1. Crea un bot en Telegram usando BotFather y copia el token del bot.

Busca en Telegram

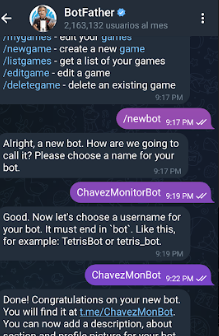


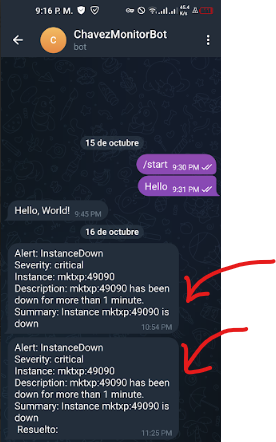
Pon es siguiente comando ” /star”, esto

dará inicio a la creación de Bot

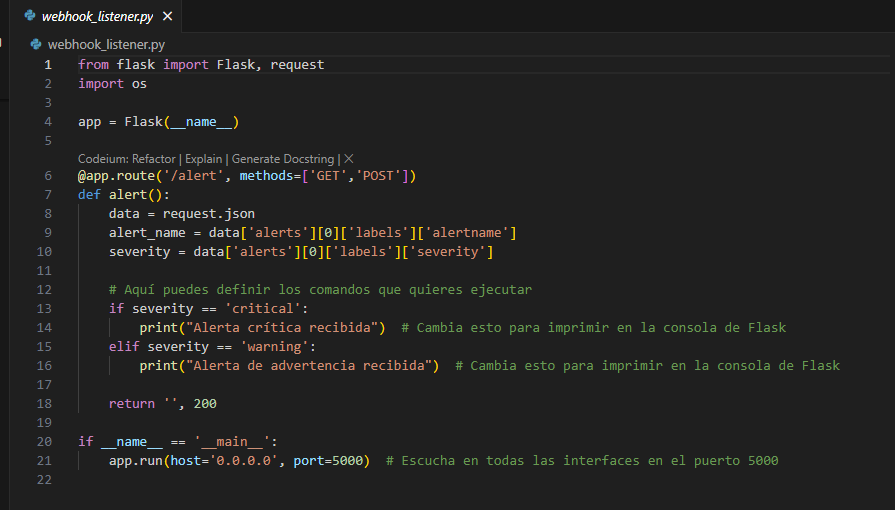


Inserta estos comandos “/newbot”, para crear un nuevo Bot, Dale un nombre en este caso yo lo nombre “ChavezMonBot”



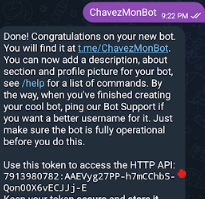
Una vez que hayas creado en “ChavezMonBot”, si tiene algún problema los Reuters, El Script reiniciara el Reuters y te mandara un mensaje con una alerta. 

Este es el script para el reinicio “webhook\_listener.py”



Para obtener el “bot token” y el “chat bot”, en telegram al insertar el nombre del

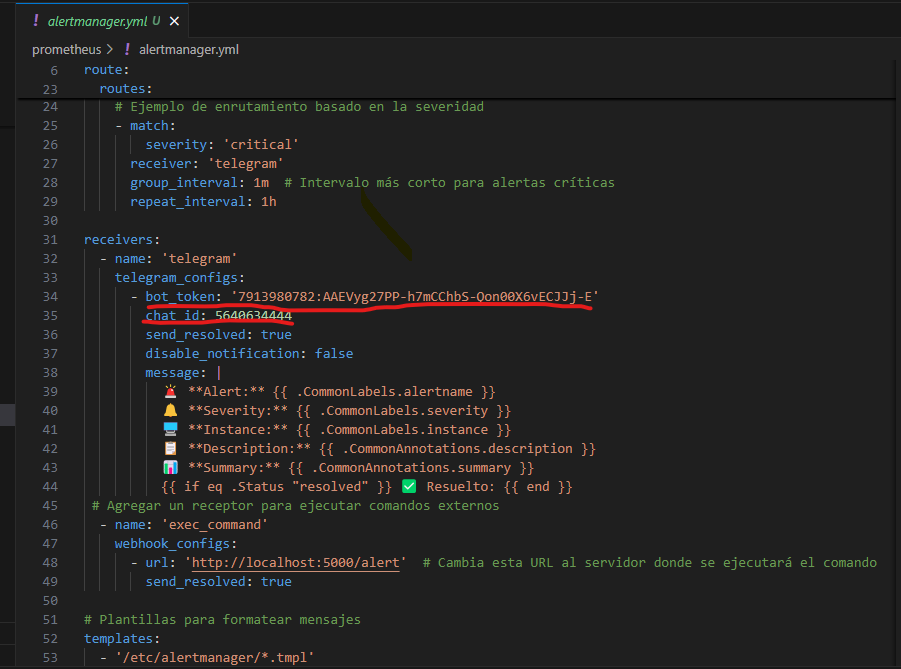
botcreado te entrega el token.



Para obtener el id escribimos en telegram “userinfobot” y escribimos el comando “/star”, y con eso obtienes el id.



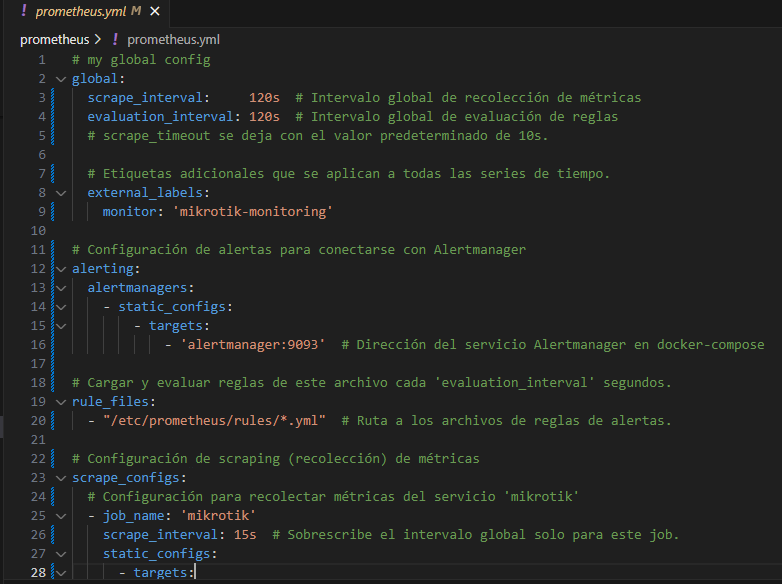
1. En el archivo “alertmanager.yml” de Alertmanager, agrega la configuración para Telegram:



1. Implementación de la API

Paso 1: Configuración de Prometheus y Grafana

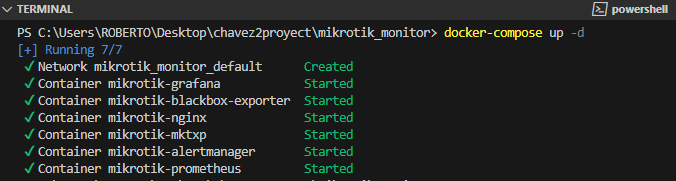
Prometheus : Configura “prometheus.yml” el directorio “prometheus” para que recoja métricas de los enrutadores.



Grafana : Asegúrese de que el tablero de Grafana esté listo grafana/provisioning/dashboards.

1. Prueba y Verificación de la API

**Paso 1: Iniciar Docker Compose**  
Ejecuta el siguiente comando en la terminal para levantar el entorno de contenedores:

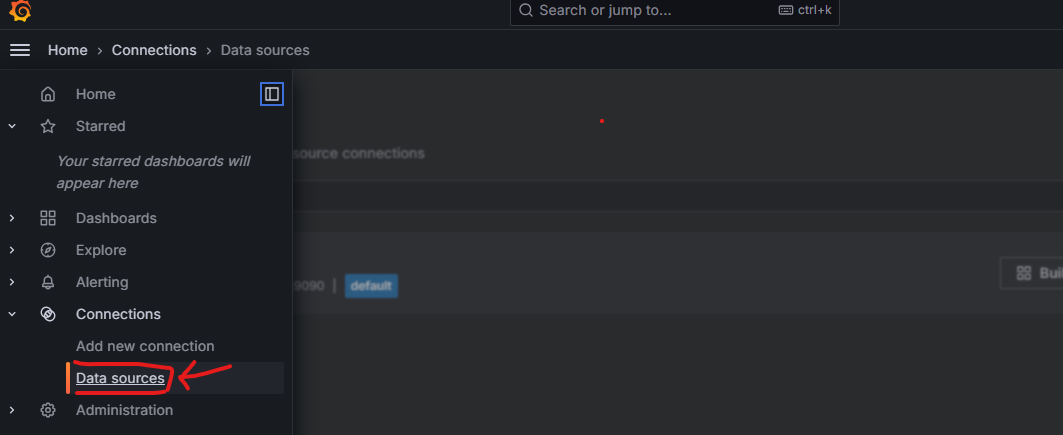


Espere unos minutos a que los servicios se inicien correctamente. Confirme el estado de los contenedores usando:

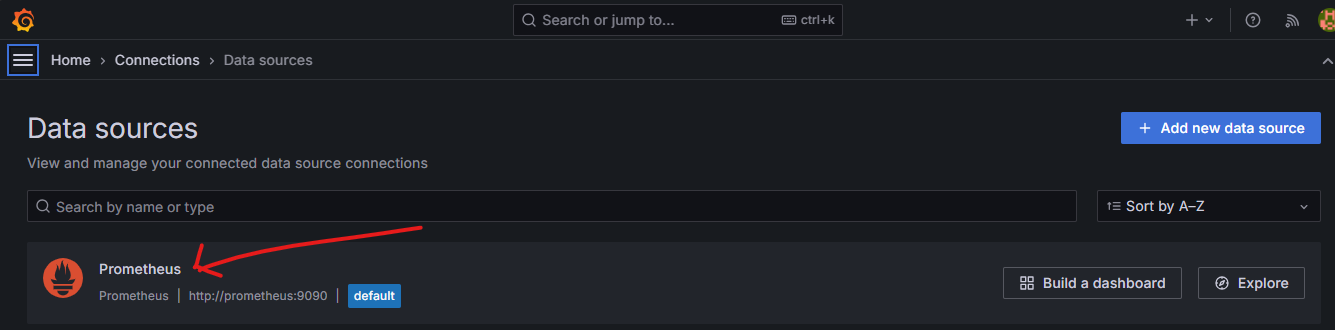
Paso 2: Acceso a Grafana y Configuración del Dashboard(en el caso de volver a configuarar y cargar el panel)

Abre Grafana en http://localhost:3000. Inicia sesión (credenciales predeterminadas: admin/admin). Si abres por primera vez. Coloca una nueva contraseña en este caso yo puse “chavez”.

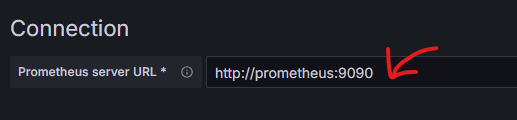
Importa el tablero de Mikrotik



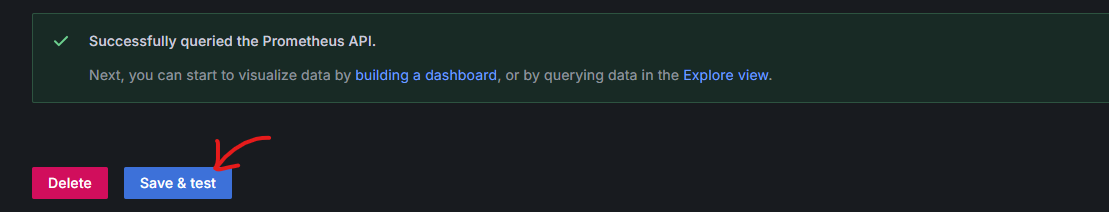
Has clic



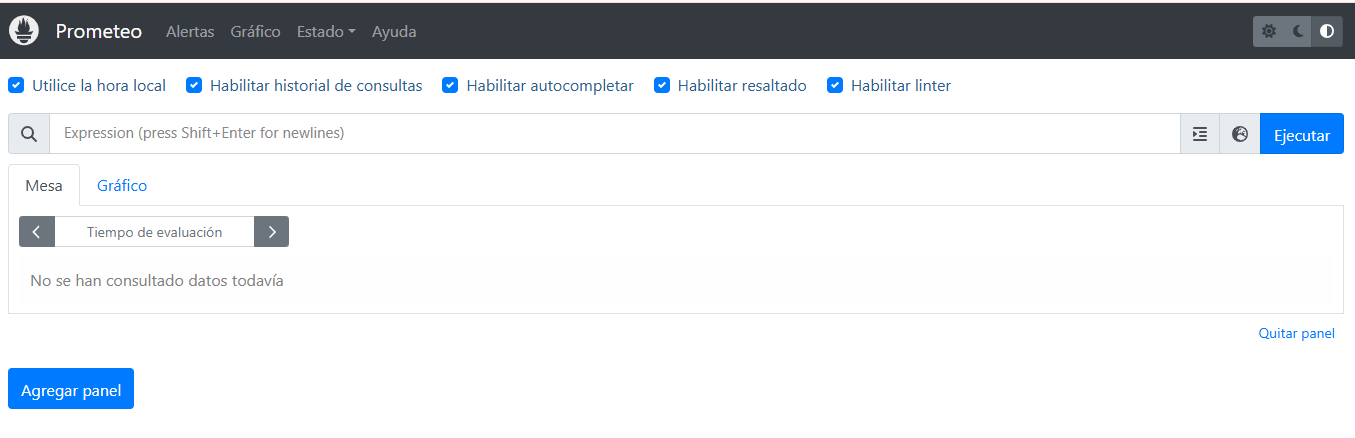
Pon la dirección



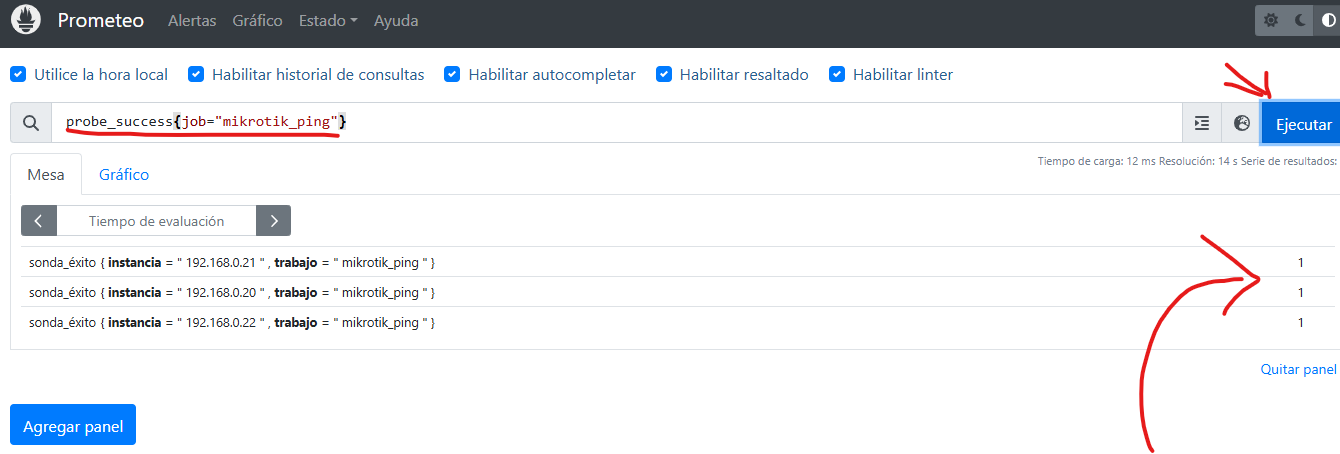
Luego guarda



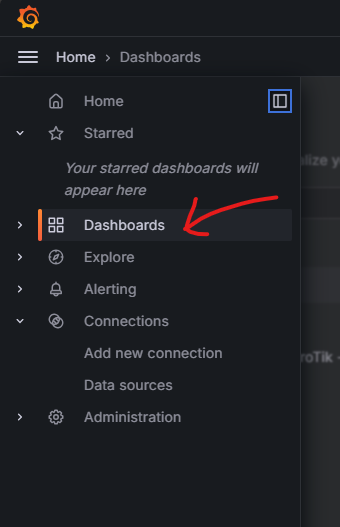
Antes tines que probar el servidor Prometheus que este corriendo los Routhers Mikrotics. “<http://localhost:9090/graph>”.



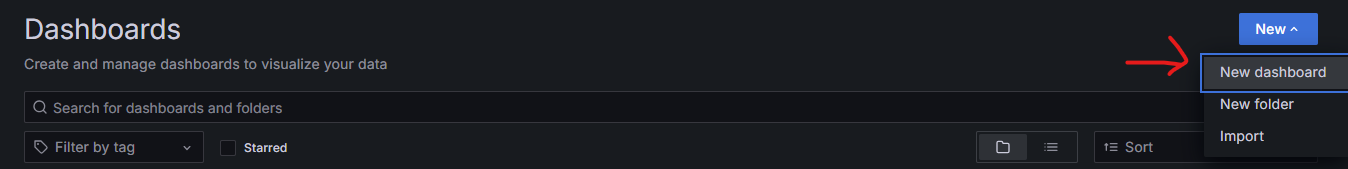
Incresa la siguiente consulta para ver si los Reuter de Mikrotick están funcionanado si están funcionando aparecerá en “**1**” y si no en “**0**”.

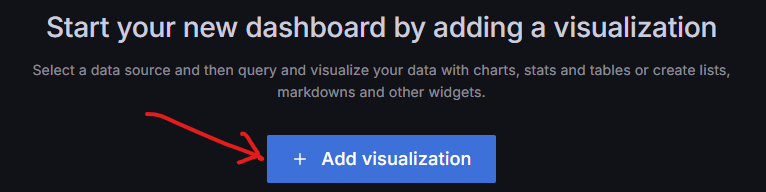


Vuelve otra vez a Grafana, Puedes crear paneles nuevos



Has click en “New dashboard” y luego + Add Visualization

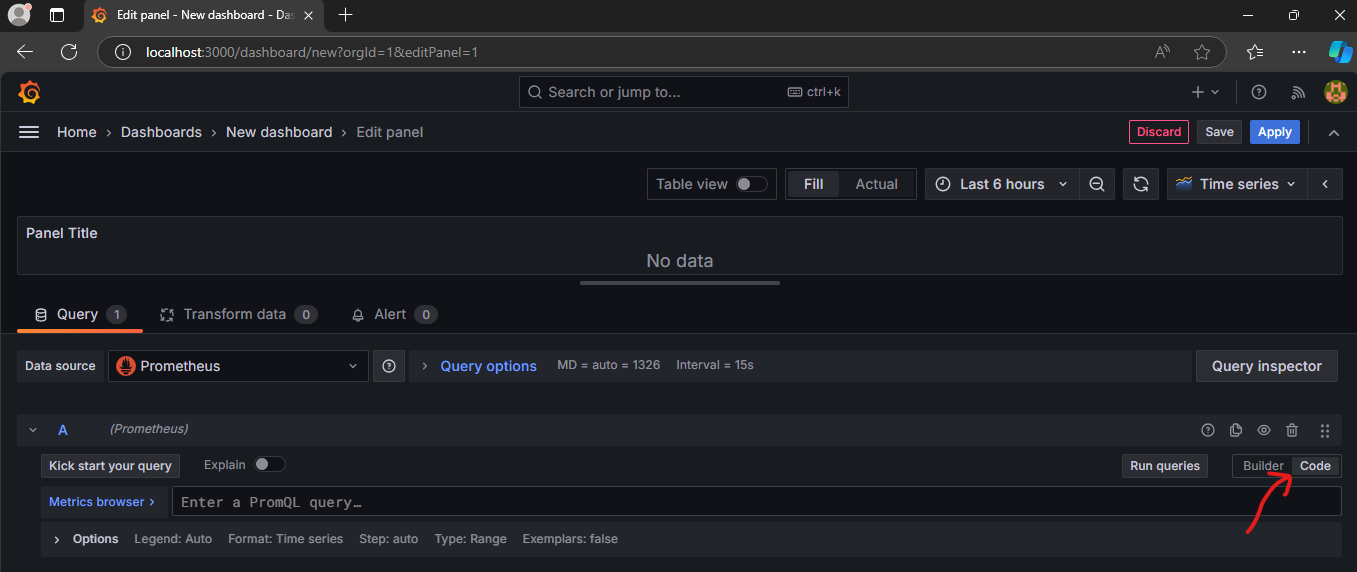




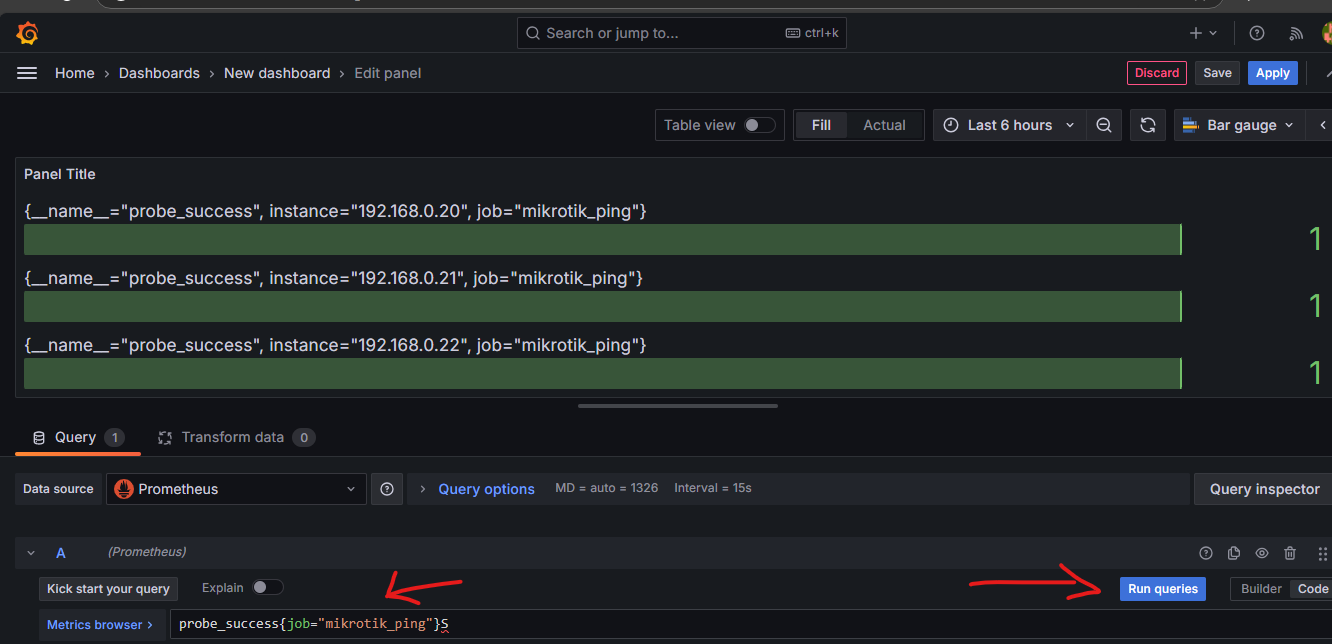
Escoges “Prometheus“



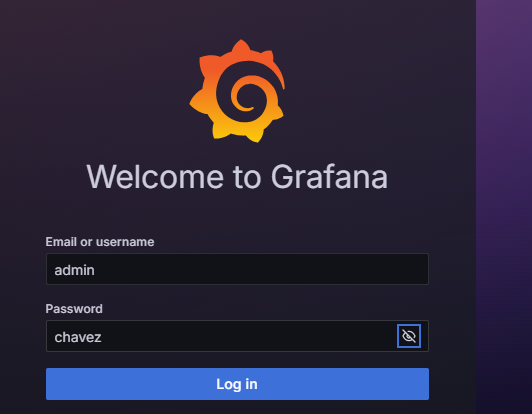
Preciona en la opción Code y ingresa la siguiente consulta “probe\_success{job="mikrotik\_ping"}”



Puedes crear miles de paneles atractivos como tú quieras. Ejemplo:



Paso 3: En el caso que no te pida una nueva contraseña prueba con esto, donde ta estaba creadas los panes de control de los Routers.

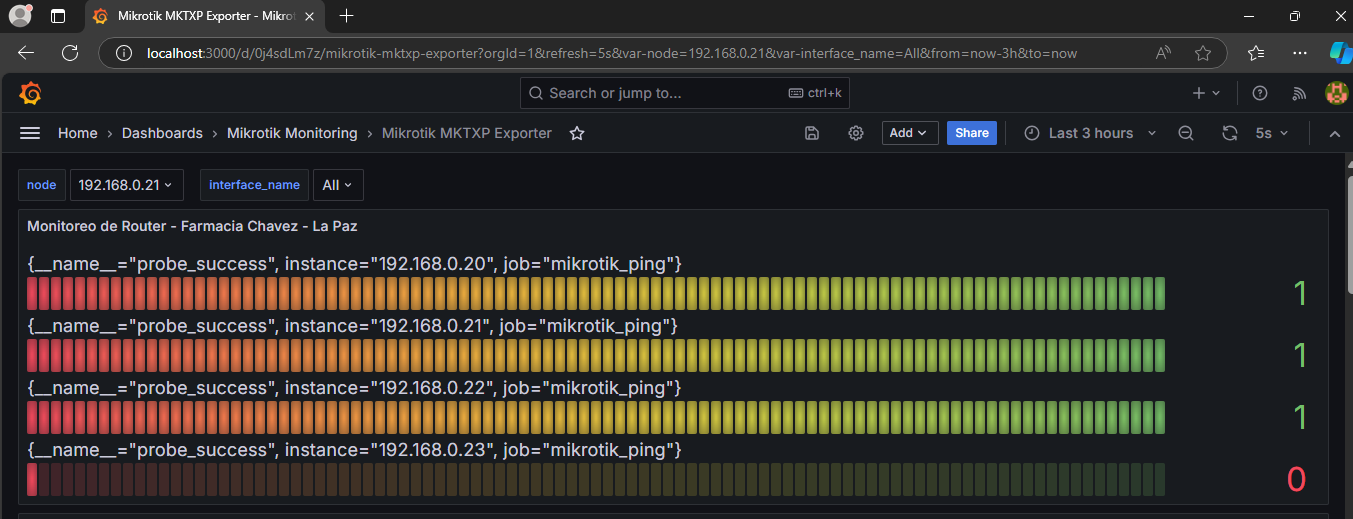


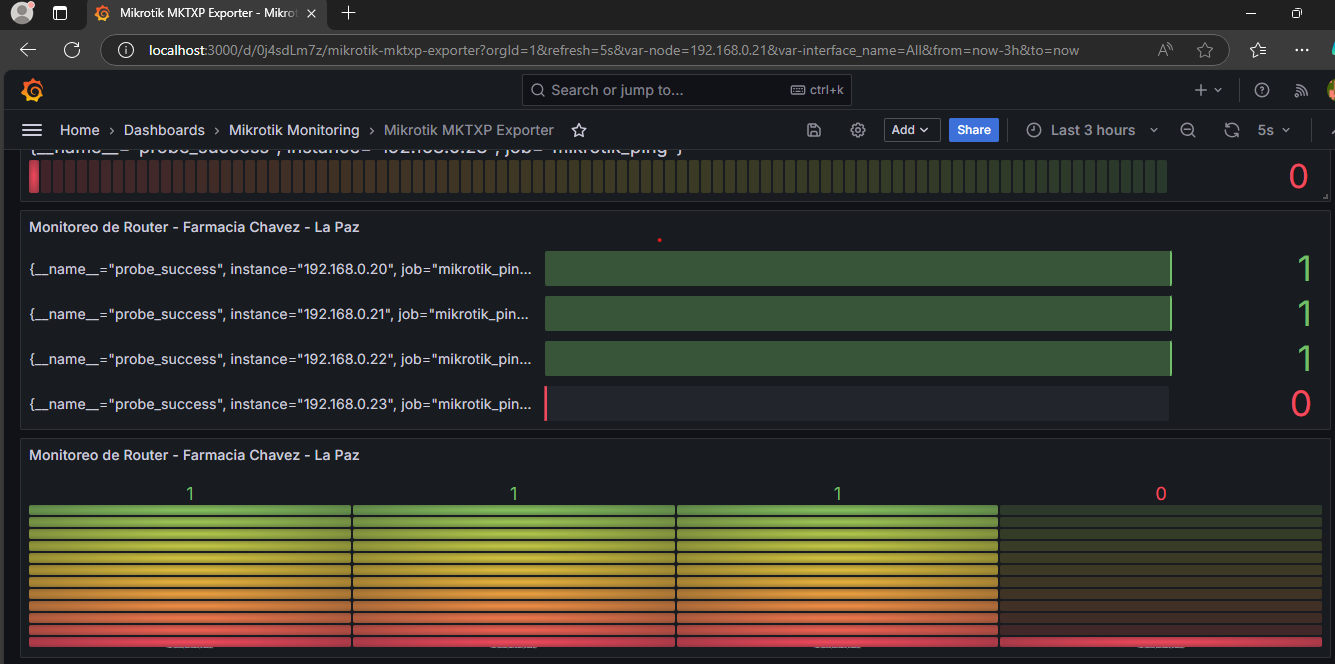
Paneles en funcionamiento “[Monitoreo de Routers MikroTik - Farmacias Chávez - Dashboards - Grafana](http://localhost:3000/d/fe26zup4gy29sf/monitoreo-de-routers-mikrotik-farmacias-chavez?orgId=1)” o ingresa en



Paneles creados con grafana por defecto solo hay tres, pero puedes aumentar la cantidad que quiera para cada Sucursal-La Paz

Hay tres Reuter funcionando “**1**” y uno caído “**0**”.

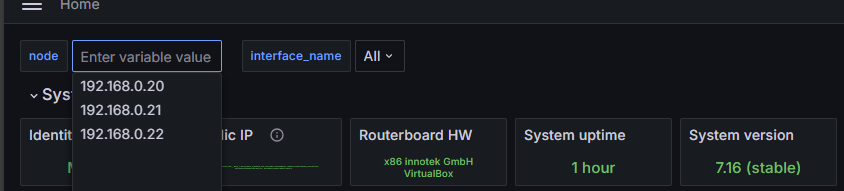


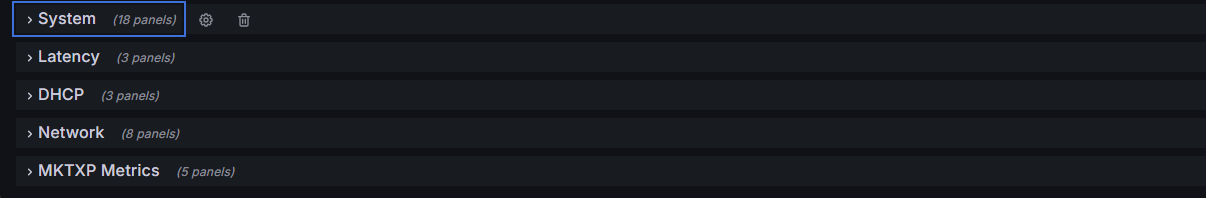




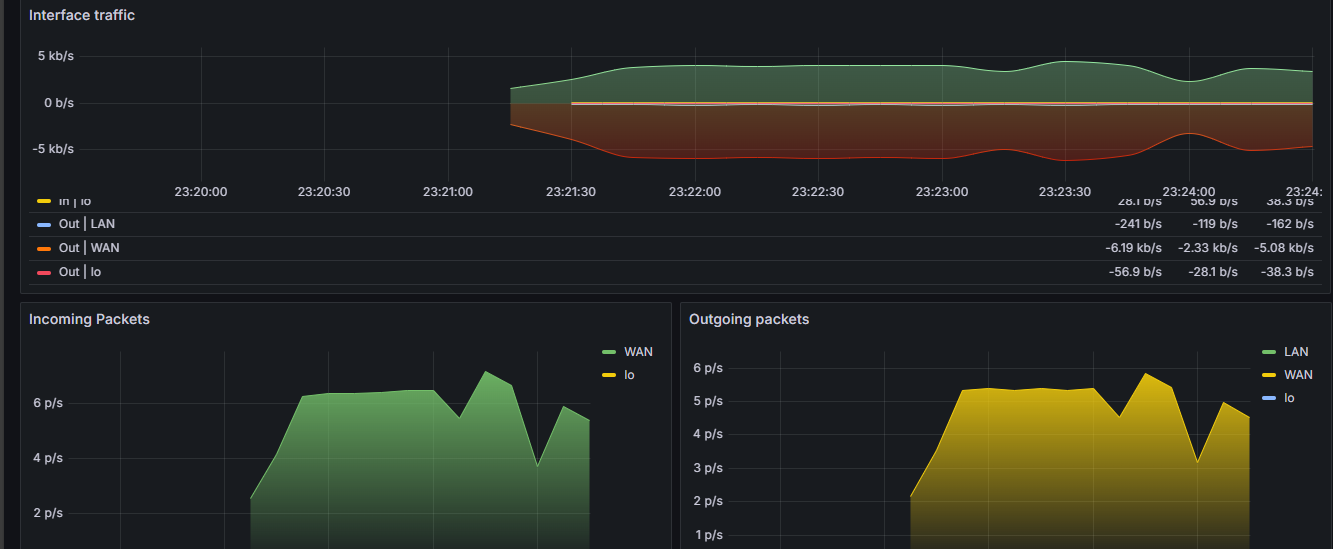


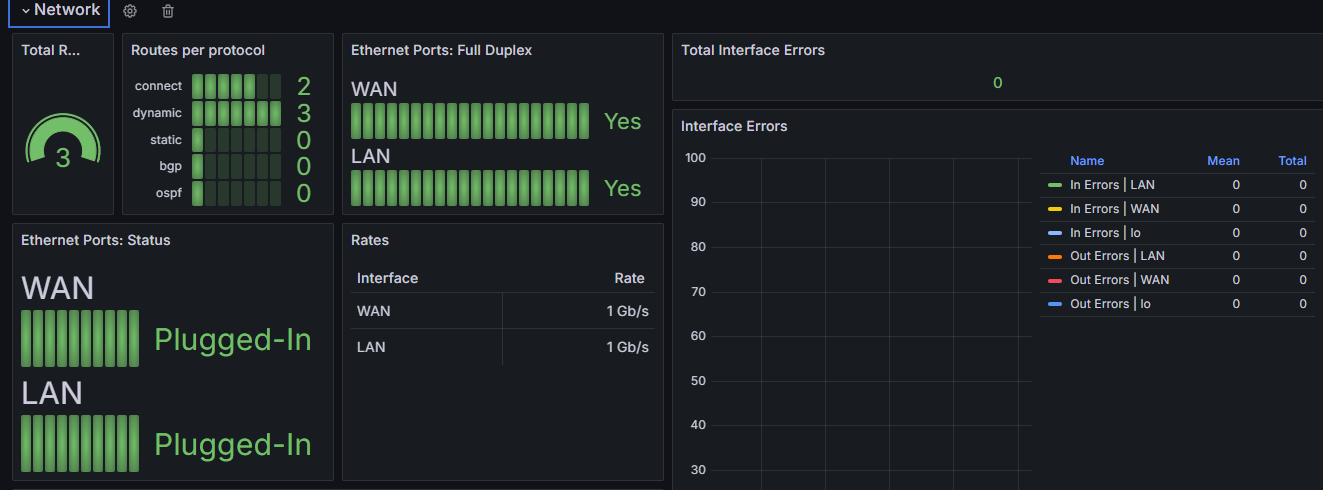
Tenemos un menú amplio ya que solo no muestra los paneles y su funcionamiento también características como: Sistema (ip, placa de enrutamiento, versión, usuarios activos, etc.).

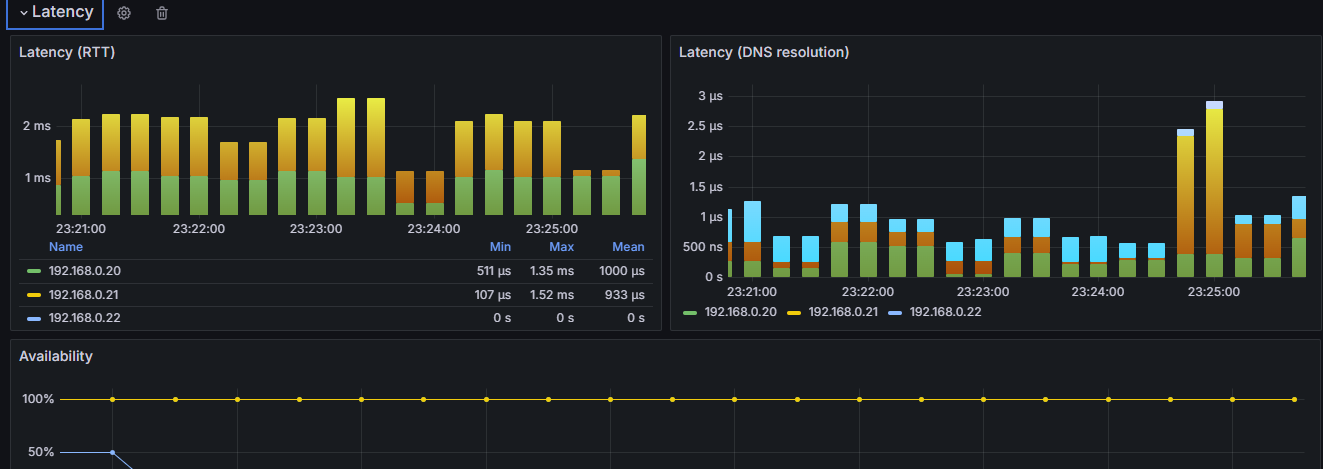


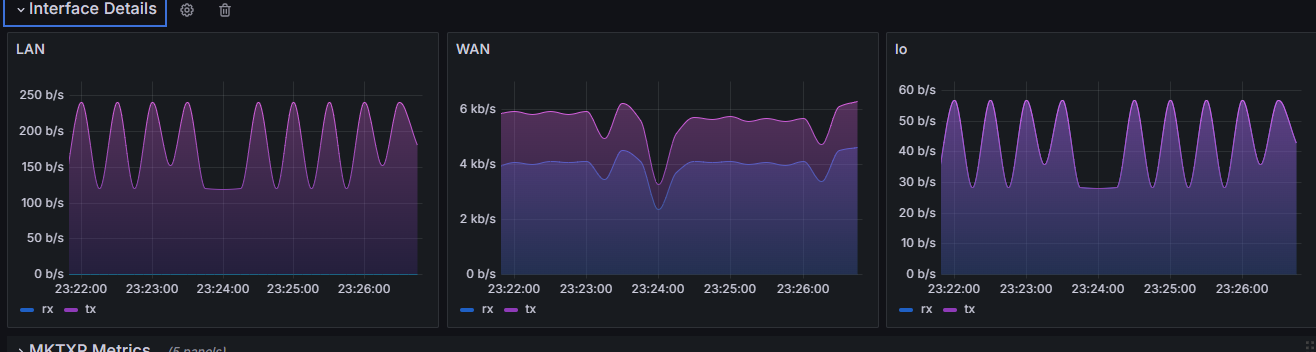


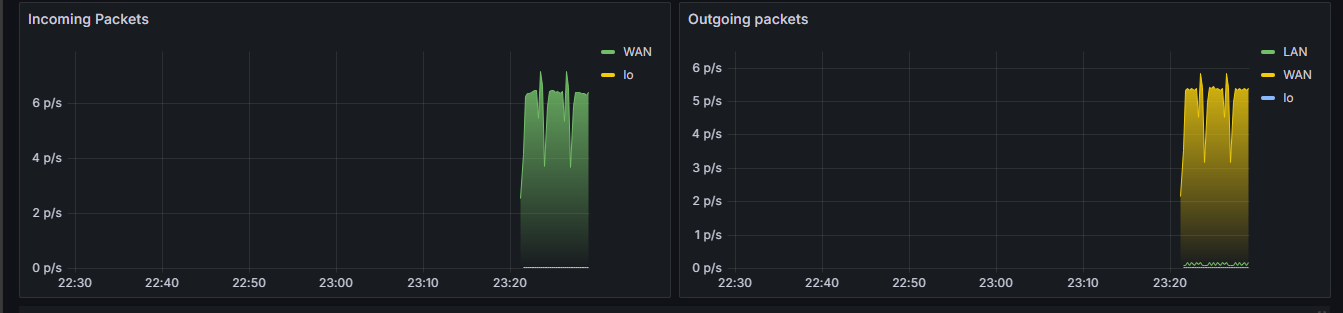












1. Funcionamiento del Sistema

* **Monitoreo de Routers con Prometheus:** Prometheus recolecta y almacenas métricas de los routers configurados en prometheus.yml. Los routers responden con métricas de estado y rendimiento, que Prometheus registra en intervalos definidos.
* **Configuración de Alertas con Prometheus y Alertmanager**: Al detectar una métrica fuera de los valores normales (como la desconexión de un router), Prometheus envía una alerta a Alertmanager, que se configura para enviar una notificación a Telegram.
* **Visualización en Grafana**: Grafana muestra visualizaciones en tiempo real de las métricas recolectadas. Puedes configurar alertas visuales en el dashboard para tener una vista rápida del estado de los routers.
* **Automatización de Reinicios**: Utiliza un script Python que detecte cambios en las métricas de conectividad y envíe comandos de reinicio a los routers mediante SSH o API de MikroTik, si una métrica indica una falla de conexión.

1. Solución de Problemas Comunes

**Prometheus no recolecta métricas**:

* Verifica que las direcciones IP y los puertos en prometheus.yml sean correctos.
* Asegúrate de que los routers estén configurados para permitir la recolección de métricas

**Alertmanager no envía notificaciones**:

Confirma que el token de Telegram y el ID del chat sean correctos.

* Revisa los logs de Alertmanager para identificar cualquier error.

**Problemas de conexión en Grafana**:

* Verifica que los contenedores estén activos (docker ps).
* Comprueba que el puerto 3000 esté libre en tu máquina.

1. Funcionalidades Adicionales

* **Dashboards de Grafana Personalizados**: Puedes crear dashboards personalizados para cada sucursal de Farmacia Chávez.
* **Logs Detallados de Alertas**: Configura Alertmanager para registrar alertas en un archivo de log para análisis posterior.
* **Notificaciones a otros Canales**: Amplía las alertas a correo electrónico o SMS si Telegram no es suficiente.