

# **Análisis de Incidencias y KPIs de Red mediante Power BI en Entornos Virtualizados con GNS3: Un Enfoque Integrado Basado en Nagios y Syslog**

Presentado por:

Carla Utrera Calderón

**Curso:** 2024/2025

**Fecha:** 25/02/2025 – 28/05/2025

## Resumen

El presente proyecto aborda la creciente necesidad de análisis eficiente de incidencias y KPIs de red en el ámbito de la consultoría, donde las empresas buscan presentar datos y rendimiento a sus clientes de manera clara y efectiva. Aprovechamos Power BI, una herramienta líder en visualización de datos, para este propósito.

Nuestro enfoque se centra en la utilización de datos obtenidos de un entorno virtualizado en GNS3. Para la recopilación de datos, se emplean diversas herramientas como Nagios y un servidor Syslog, permitiendo la captura de tráfico de red, el monitoreo de recursos y la recopilación de registros del sistema.

Los datos obtenidos son procesados en Power BI, donde se realizan transformaciones, modelado y la creación de medidas con DAX para generar dashboards interactivos. Estas visualizaciones permiten evaluar el desempeño de la red en términos de incidencias, tiempos de resolución, disponibilidad y rendimiento del ancho de banda.

El proyecto responde a diversas necesidades del sector, como la ausencia de una herramienta centralizada para el análisis de redes, la baja satisfacción en la detección y resolución de incidencias, las dificultades para cumplir estándares de calidad y el impacto de los cambios tecnológicos en la administración de redes.

Finalmente, se implementa un caso práctico en un entorno real para validar la metodología, presentando conclusiones, limitaciones y propuestas de mejora para futuros estudios.

## TABLA DE CONTENIDO

|  |    |
|--|----|
| Resumen.....   | 2  |
| Índice de figuras.....                                     | 6  |
| FASE 1: Diseño del Proyecto.....                           | 8  |
| 1.1 Generación de ideas del proyecto .....                 | 8  |
| 1.2 Justificación de la elección del proyecto .....        | 8  |
| 1.3 Definición del proyecto.....                           | 8  |
| 1.4 Definición de los objetivos .....                      | 9  |
| 1.5 Estado del arte.....                                   | 9  |
| FASE 2: Planificación del Proyecto.....                    | 12 |
| 2.1 Planificación de actividades.....                      | 12 |
| 2.2 Diagrama de Gantt.....                                 | 14 |
| 2.3 Presupuesto detallado del proyecto.....                | 15 |
| 2.5. Análisis de riesgos .....                             | 16 |
| FASE 3: Ejecución del Proyecto .....                       | 18 |
| 3.1. Descripción de las tareas realizadas .....            | 18 |
| 3.2. Arquitectura de máquinas virtuales.....               | 18 |
| 3.2.2 Elección del hipervisor .....                        | 19 |
| 3.2.3 Requisitos de hardware recomendados.....             | 20 |
| 3.2.4 GNS3 VM: descripción y ventajas.....                 | 20 |
| 3.3. Introducción general .....                            | 21 |
| 3.3.1 Componentes principales .....                        | 22 |
| 3.3.2 Interacción según el modelo OSI .....                | 22 |
| 3.4 Instalación de los requisitos previos .....            | 23 |
| 3.5 Configuración de GNS3 para uso con GNS3 VM Server..... | 23 |
| 3.5.1 Funcionamiento y distribución de carga en GNS3 ..... | 27 |
| 3.6 Configuración detallada de VirtualBox .....            | 27 |
| 3.6.1 Máquina virtual para Servidor Nagios .....           | 27 |
| 3.6.2 Máquina virtual para Servidor Syslog .....           | 28 |
| 3.6.3 Máquinas virtuales para usuarios (PC1 y PC2) .....   | 29 |
| 3.7 Configuración de red con IP estática .....             | 29 |
| 3.8 Añadir las VMs a GNS3 .....                            | 30 |
| 3.9 Configurar GNS3 y añadir los dispositivos.....         | 33 |
| 3.9.1 Añadir y configurar dispositivos en GNS3.....        | 33 |

|   |    |
|---|----|
| 3.9.2 Añadir los routers al proyecto .....  | 39 |
| 3.9.3 Uso de un Switch Cisco Catalyst de Capa 3.....  | 39 |
| 3.9.4 Añadir el switch al proyecto .....  | 44 |
| 3.10 Pasos para añadir una Cloud en GNS3 .....  | 44 |
| 3.11 Conexión física de la topología.....   | 46 |
| 3.13 Conexión de la Topología de GNS3 a Internet mediante Interfaz Loopback y Adaptador Bridge..... | 46 |
| 3.13.1 Creación y configuración de la interfaz Loopback .....                                       | 46 |
| 3.13.2 Compartir Internet con GNS3 .....  | 47 |
| 3.13.3 Configuración en GNS3: Añadir Cloud y vincular interfaz Loopback .....                       | 48 |
| 3.13.4 Configuración del router en GNS3.....  | 50 |
| 3.13.5 Verificación de conectividad .....   | 50 |
| 3.14 Configuración Server Syslog .....  | 50 |
| 3.14.1 Instalación y Activación del Servicio Syslog .....   | 51 |
| 3.14.2 Configurar recepción de logs remotos .....   | 52 |
| 3.14.3 Editar configuración de rsyslog para escuchar por red .....                                  | 52 |
| 3.14.4 Configurar Firewall.....   | 53 |
| 3.14.5 Reiniciar rsyslog para aplicar cambios.....  | 53 |
| 3.14.6 Configuración de Cliente (PCs Linux) .....   | 53 |
| 3.14.7 Verificar persistencia tras reinicio .....   | 54 |
| 3.14.8 Configuración en Routers Cisco .....   | 54 |
| 3.14.9 Verificación en el Servidor Syslog.....  | 55 |
| 3.14.10 Generar Logs para Pruebas .....   | 55 |
| 3.15 Copiar los datos a mi pc mediante un túnel SSH reverse .....                                   | 56 |
| 3.15.1 Pasos para exportar logs desde Ubuntu-Server a Windows.....                                  | 57 |
| 3.16 Configuración Server Nagios .....  | 60 |
| 3.16.1 Preparación del sistema .....  | 61 |
| 3.16.2 Instalar Nagios Core .....   | 61 |
| 3.16.3 Instalar plugins.....  | 62 |
| 3.16.4 Acceso Web.....  | 62 |
| 3.16.5 Configuración personalizada.....   | 62 |
| 3.16.6 Acceder a la interfaz web de Nagios.....   | 70 |
| 3.17 Acceso al panel de Nagios y la extracción de datos .....                                       | 71 |

|  |     |
|--|-----|
| 3.17.1 Port-forwarding para acceder a Nagios desde el exterior .....               | 71  |
| 3.17 Extracción de los datos del Servidor Nagios.....                              | 75  |
| 3.18 Marco Teórico de Power BI .....   | 76  |
| 3.19 Diseño e Implementación del Sistema de Análisis .....                         | 77  |
| 3.20 Modelado de datos .....   | 81  |
| 3.21 Creación de medidas y KPIs en DAX y Desarrollo de Dashboards y Reportes ..... | 87  |
| 3.21.1 Incidencias del sistema (tabla syslog_events) .....                         | 89  |
| 3.21.2 Alertas de infraestructura (tabla nagios_alerts).....                       | 90  |
| 3.21.3 Incidencias detalladas (tabla nagios_eventlog) .....                        | 91  |
| FASE 4: Resultados y evaluación .....  | 95  |
| 4.1 Conclusiones y Trabajo Futuro .....  | 95  |
| 4.2 Trabajo Futuro .....   | 95  |
| Bibliografía .....   | 96  |
| Anexos .....   | 97  |
| A. Código SQL utilizado.....   | 97  |
| B. Código DAX utilizado.....   | 99  |
| C. Configuración lógica de la red.....   | 102 |
| Glosario de términos .....   | 107 |

## Índice de figuras

|  |    |
|--|----|
| Imagen 1 Diagrama de Gantt.....  | 14 |
| Imagen 2 Análisis de Riesgos .....   | 16 |
| Imagen 3 Arquitectura de la virtualización.....  | 19 |
| Imagen 4 Prototipo de la virtualización en Gns3 .....  | 21 |
| Imagen 5 Virtualización de GNS3 VM .....   | 26 |
| Imagen 6 Vista del Centro de redes y recursos compartidos mostrando la red Wi-Fi principal y la red virtual “GNS3” con acceso a Internet.....    | 47 |
| Imagen 7 Propiedades del adaptador Wi-Fi mostrando el uso compartido activado y vinculado a la red virtual GNS3.....                             | 48 |
| Imagen 8 Conexión a Internet.....  | 49 |
| Imagen 9 Estado y configuraConexión a Internetción de la interfaz “GNS3”, con IP estática 192.168.137.1 y DNS públicos de Google (8.8.8.8). .... | 50 |
| Imagen 10 Funcionamiento de Syslog .....   | 51 |
| Imagen 11 Servicio activo de Syslog Server .....   | 52 |
| Imagen 12 Configuración de Cliente PC1 (Syslog) .....  | 54 |
| Imagen 13 Logs del Syslog.....   | 55 |
| Imagen 14 Túnel SSH reverso.....   | 57 |
| Imagen 15 NAT para exponer el puerto 22 del Ubuntu-Server (10.10.11.2) al exterior a través del puerto 2222 .....                                | 57 |
| Imagen 16 Conexión del túnel establecida con el servidor.....  | 58 |
| Imagen 17 Arquitectura de Nagios.....  | 60 |
| Imagen 18 Servicio activo de Nagios Server.....  | 63 |
| Imagen 19 Correcta configuración de los archivos Nagios definidos.....   | 70 |
| Imagen 20 Descarga del directorio var de Nagios.....   | 75 |
| Imagen 21 Esquema de funcionamiento de Power Bi .....  | 76 |
| Imagen 22 Modelo relacional de la base de datos .....  | 78 |
| Imagen 23 Configuración de la contraseña del servidor local .....  | 79 |
| Imagen 24 Acceso mediante Heidi al entorno MySQL .....   | 79 |
| Imagen 25 Estructura SQL.....  | 80 |
| Imagen 26 Base de datos tfg .....  | 80 |
| Imagen 27 Datos de tfg.....  | 80 |



## FASE 1: Diseño del Proyecto

### 1.1 Generación de ideas del proyecto

Esta fase se centra en la conceptualización y definición de los objetivos del proyecto. Se buscará identificar las necesidades o problemas que se desean abordar, generar ideas para resolverlos y establecer prioridades entre las posibles soluciones.

La fase también incluye la descripción del proyecto y la definición de sus objetivos generales y específicos, lo que permitirá establecer una base sólida para la planificación y ejecución del proyecto en fases posteriores.

### 1.2 Justificación de la elección del proyecto

A nivel empresarial, he constatado que uno de los aspectos clave para gestionar proyectos de IT orientados a los clientes es el análisis exhaustivo de los servicios que la empresa ofrece, así como la eficacia y eficiencia en su rendimiento. Las organizaciones demandan herramientas que permitan presentar datos y métricas de rendimiento de manera clara y precisa, facilitando así la toma de decisiones estratégicas.

En el ámbito de la consultoría, Power BI se destaca como una herramienta altamente reconocida por su capacidad para transformar datos en visualizaciones interactivas y dashboards intuitivos. Esto permite mostrar indicadores clave de rendimiento (KPIs) y analizar incidencias de manera eficaz, lo cual me llamó especialmente la atención para la elección de este proyecto.

Personalmente, he trabajado con Power BI para el análisis de datos, lo que me ha permitido apreciar de primera mano el potencial de esta herramienta. Además, tecnologías complementarias como Nagios, que se utilizan en la empresa para la monitorización, refuerzan el valor añadido de una solución integral.

Por ello, este proyecto no solo resulta técnicamente relevante, sino que también aborda un aspecto fundamental en la consultoría de IT.

No obstante, mi elección también se debe a que los proyectos enfocados en el mantenimiento de una red pasan desapercibidos, siendo los principales aquellos centrados en la implementación.

### 1.3 Definición del proyecto

Este proyecto se centra en el procesamiento y análisis de datos en Power BI, utilizando información obtenida de un entorno virtualizado en GNS3. Se integran diversas herramientas, como Nagios para el monitoreo de recursos y un servidor Syslog para la recopilación de registros del sistema.

El objetivo es transformar estos datos en información a través del modelado de datos y la creación de medidas utilizando DAX (Data Analysis Expressions), lo que permite desarrollar dashboards interactivos y dinámicos. Estos paneles de

control facilitarán la visualización en tiempo real de indicadores clave (KPIs) y el análisis de incidencias, proporcionando a los usuarios una herramienta robusta para la toma de decisiones estratégicas y la optimización del rendimiento de la red.

#### **1.4 Definición de los objetivos**

El objetivo principal de este proyecto es resaltar la importancia de analizar los recursos de una red tras su implementación, asegurando un funcionamiento óptimo, ya que es crucial mantener la infraestructura en buen estado.

Con este enfoque, las organizaciones podrán presentar de manera clara y precisa datos y métricas de rendimiento que permitan cumplir con los estándares de calidad del servicio contratado.

Para llevar a cabo este proyecto, realizaré una recopilación de datos en formato CSV de los dispositivos que forman parte de un entorno virtual de GNS3 en mi PC. Estos datos se usarán en Power BI para crear dashboards y KPIs que permitan monitorear el rendimiento de la red.

Gracias a Power BI, podré generar un informe técnico que compile toda la información de mi red virtualizada, permitiendo evaluar su rendimiento y generar métricas clave para mejorar la calidad del servicio de red.

En resumen, este enfoque ofrece ventajas significativas al proporcionar un entorno de aprendizaje eficiente para estudiantes y/o empleados que deseen simplificar la gestión de recursos informáticos, mejorar la seguridad y optimizar el acceso a aplicaciones y herramientas educativas.

Un ejemplo práctico de su aplicación sería la creación de informes mensuales que se presenten a los clientes para evaluar el rendimiento del servicio.

#### **1.5 Estado del arte**

Este estado del arte ofrece una visión general de los avances más relevantes en cuanto a herramientas de visualización de datos, monitorización de redes y el uso de KPIs en la gestión de redes informáticas.

**Monitoreo de Redes y Gestión de Incidencias:** La administración de redes se ha visto cada vez más impulsada por el uso de herramientas que proporcionan datos detallados sobre el estado de los recursos y el tráfico en la infraestructura. Entre las herramientas más utilizadas se encuentran:

- **Nagios:** Esta herramienta se destaca en la monitorización de infraestructuras de TI, proporcionando capacidades para observar el estado de los dispositivos y recursos de la red, como servidores, routers, switches, etc. Nagios facilita la detección temprana de problemas en la red y la generación de alertas cuando un sistema falla o no cumple con los parámetros establecidos. Es ampliamente utilizado en grandes entornos empresariales debido a su flexibilidad y capacidad de integración con otros sistemas de TI.

- **Syslog:** Es un estándar utilizado para la recopilación de registros de eventos de dispositivos de red. A través de servidores Syslog, los dispositivos envían sus logs, lo que permite centralizar la información sobre el comportamiento y estado de la red. Este tipo de análisis es clave para diagnosticar problemas relacionados con la red, como fallos en los sistemas o intentos de acceso no autorizado.

**Power BI como Herramienta de Business Intelligence en Redes** Power BI es una herramienta de visualización de datos de Microsoft que permite transformar grandes volúmenes de información en informes interactivos y visuales. Se ha convertido en un estándar en el ámbito de Business Intelligence (BI), debido a su facilidad de uso, capacidad de integración con múltiples fuentes de datos y su poderosa capacidad para generar dashboards dinámicos y detallados. En el análisis de redes, Power BI se utiliza para procesar datos de diversas fuentes (como los mencionados Nagios y Syslog), transformándolos en KPIs y métricas visuales de fácil interpretación.

- **Creación de Dashboards Interactivos:** Power BI permite la creación de paneles de control que representan visualmente el estado de la red, donde se pueden incluir indicadores como la disponibilidad de la red, el rendimiento del ancho de banda, el tiempo de resolución de incidencias y la frecuencia de los eventos de red. Estos dashboards interactivos ofrecen una representación clara del rendimiento de la infraestructura, lo que facilita la toma de decisiones.
- **DAX (Data Analysis Expressions):** Power BI utiliza DAX para crear medidas y KPIs personalizados que permiten analizar y comparar datos en función de parámetros específicos. Por ejemplo, la creación de indicadores como el tiempo medio de resolución de incidencias o la disponibilidad de red son vitales para la gestión de redes y pueden ser calculados fácilmente mediante DAX.

**Integración con otras herramientas de monitoreo:** Power BI también se integra de manera eficiente con herramientas como Nagios y servidores Syslog, lo que permite la centralización de datos en un único lugar. Esto simplifica la visualización y el análisis, y facilita la creación de informes técnicos y presentaciones para los clientes.

**KPIs en Redes Informáticas** Los KPIs (Indicadores Clave de Rendimiento) son fundamentales en la gestión de redes, ya que permiten medir el desempeño de la infraestructura y la calidad del servicio. En el contexto de las redes, los KPIs más comunes incluyen:

- **Disponibilidad de la Red:** Mide el tiempo en que la red está operativa sin fallos. Un KPI crítico para garantizar la continuidad de las operaciones y la calidad del servicio.
- **Tiempo Medio de Resolución de Incidencias:** Mide el tiempo promedio que se tarda en resolver un problema desde que es identificado hasta que se soluciona. Este KPI ayuda a evaluar la eficiencia del equipo de soporte y la efectividad en la gestión de incidencias.

- **Rendimiento del Ancho de Banda:** Mide la cantidad de datos que pueden ser transmitidos a través de la red en un tiempo determinado. Un KPI clave para asegurar que la red no está sobrecargada y puede manejar el volumen de tráfico esperado.

**Tendencias Actuales y Futuras** En los últimos años, las soluciones de análisis de redes se han estado moviendo hacia la integración de tecnologías emergentes como la inteligencia artificial (IA) y el aprendizaje automático (ML) para mejorar la detección de incidencias y prever problemas antes de que ocurran. Estas tecnologías están empezando a ser utilizadas para analizar patrones de tráfico y comportamiento de red, lo que puede ayudar a prevenir incidentes y optimizar la red de manera proactiva.

Además, la virtualización y las redes definidas por software (SDN) están remodelando la infraestructura de redes, lo que hace que el análisis de redes y el monitoreo de incidencias sea aún más relevante, ya que los recursos son más dinámicos y distribuidos.

**Retos en la Implementación de Herramientas de Análisis de Redes** Aunque existen herramientas poderosas para la monitorización y el análisis de redes, la implementación y el uso efectivo de estas herramientas enfrenta algunos desafíos:

- **Integración de Herramientas:** Integrar diversas fuentes de datos (como Nagios y Syslog) en una plataforma unificada como Power BI puede ser complejo, ya que los formatos de datos y las interfaces de integración pueden variar.
- **Escalabilidad:** Las redes grandes o complejas requieren soluciones de monitoreo que puedan escalar adecuadamente, lo que presenta retos tanto a nivel de hardware como de software.
- **Interpretación de los Datos:** A pesar de la capacidad de Power BI para procesar y visualizar grandes volúmenes de datos, interpretar correctamente los resultados y tomar decisiones basadas en estos puede requerir experiencia técnica y conocimiento profundo de la infraestructura de la red.

## FASE 2: Planificación del Proyecto

### 2.1 Planificación de actividades

La **primera fase** del proyecto se centra en su puesta en marcha y valoración preliminar. Es fundamental, pues en este momento se decide si conviene avanzar con la propuesta o no. A continuación, se presentan las tareas más relevantes a llevar a cabo:

- **Estudio de la tecnología**
  - Revisión y aprendizaje continuo de GNS3, Nagios y Syslog.
  - Actualización permanente sobre nuevas funcionalidades y mejores prácticas.
- **Análisis de mercado y viabilidad**
  - Evaluación de la demanda y casos de uso reales.
  - Análisis de costes, recursos necesarios y retorno esperado.
  - Decisión de continuar o descartar la iniciativa según resultados.
- **Definición de alcance y requisitos técnicos**
  - Delimitación funcional: objetivos, KPIs de red y requisitos de monitorización.
  - Identificación de recursos: hardware, licencias, personal y plazos.
- **Instalación y configuración de entornos virtualizados con GNS3**
  - Despliegue de la infraestructura base (routers, switches, servidores).
  - Integración de máquinas virtuales para Nagios Core y servidor Syslog.
- **Configuración inicial de Nagios y Syslog**
  - Instalación de Nagios Core: parámetros básicos, plantillas de hosts y servicios.
  - Despliegue de un servidor Syslog centralizado: normalización y etiquetado de registros.

La **segunda fase** se centra en desplegar e instalar los servicios que se ofrecerán al cliente. Debido a la complejidad de este bloque, surgirán múltiples retos que habrá que resolver sobre la marcha, de modo que esta fase suele ser la más laboriosa del proyecto. A continuación, se detallan las principales actividades a realizar:

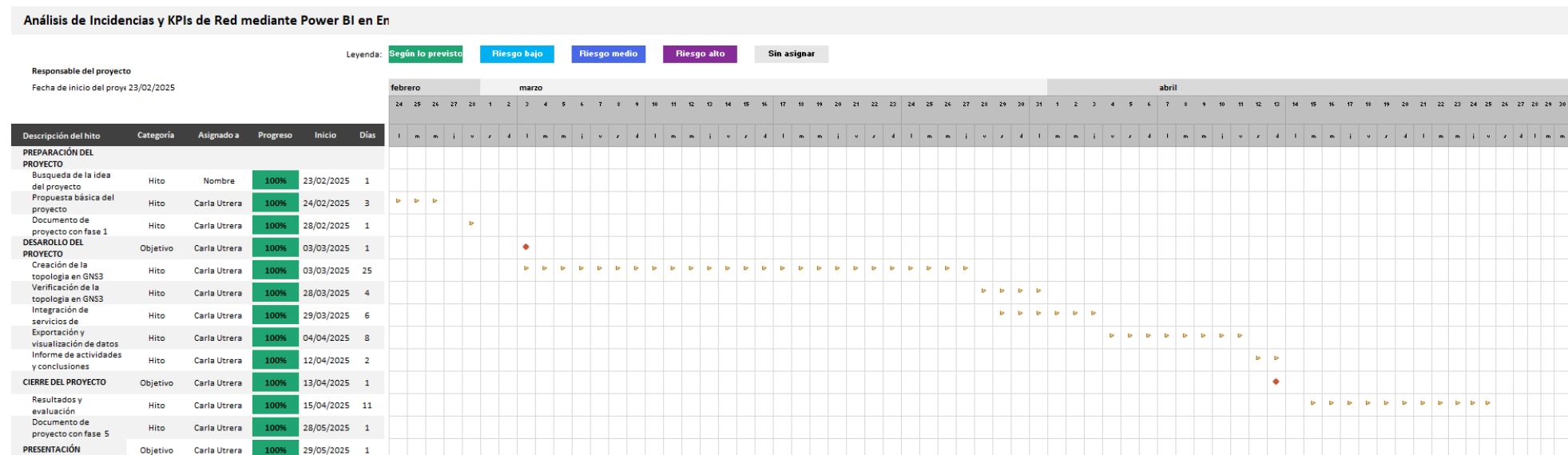
- **Parametrización avanzada de Nagios**
  - Definición de umbrales y escalados de notificaciones.
  - Creación de grupos de hosts y servicios críticos.
- **Implementación del sistema de logs con Syslog**
  - Filtrado y clasificación de eventos críticos.
  - Almacenamiento y rotación de ficheros de log.
- **Desarrollo del modelo de datos en Power BI**
  - Creación de base de datos relacional (MySQL con XAMPP y HeidiSQL).
  - Modelado semántico y conexión con Excel para definición de medidas DAX.

- **Diseño de paneles interactivos**
  - Construcción de dashboards para KPIs de rendimiento, disponibilidad y seguridad.
  - Configuración de filtros, alertas visuales y medidas personalizadas.

En la **tercera etapa** se procede al cierre del proyecto. Al ser la fase más breve, consiste principalmente en entregar el servicio, formalizar la finalización mediante la firma del albarán y dar por concluida la iniciativa. Actividad breve centrada en la entrega formal del servicio y la documentación.

## 2.2 Diagrama de Gantt

El diagrama de Gantt presentado a continuación constituye la pieza central de la planificación temporal de mi Trabajo Fin de Grado, «**Análisis de incidencias y KPIs de red mediante Power BI**». Se trata de una representación visual que desglosa el proyecto en cinco grandes fases —**Preparación, Diseño, Desarrollo, Control del proyecto y Presentación**— y coloca cada actividad en una línea de tiempo diaria para los meses de marzo y abril.



### **Imagen 1 Diagrama de Gantt**

## 2.3 Presupuesto detallado del proyecto

El presente presupuesto se ha elaborado con la expectativa de que nuestra empresa de consultoría sea contratada por el Cliente para implantar una **solución integral de virtualización, monitorización y analítica de datos.**

El alcance del proyecto comprende:

- **Suministro de hardware** para la creación de laboratorios virtualizados.
- **Provisión de licencias de software** (Power BI Pro, Windows 11 Pro y eventuales complementos de GNS3).
- **Servicios profesionales** de instalación, integración y pruebas de las herramientas Nagios, Syslog y Power BI.
- **Capacitación del personal clave** y entrega de la documentación técnica necesaria para la operación y mantenimiento de la solución.

| Concepto                             | Detalle   | Coste aprox. (EUR) |
|--------------------------------------|---|--------------------|
| <b>Hardware y equipos</b>            | PC para virtualización (Intel Core i7/i9 o AMD Ryzen, 32 GB RAM, SSD 1 TB, GPU básica)                                    | <b>1 323 €</b>     |
| <b>Software y licencias</b>          | Power BI Pro (1 usuario, anual) → 106 €<br>Windows 11 Pro → 176 €<br>Complementos GNS3 (si fueran necesarios) → 0 – 176 € | <b>282 – 458 €</b> |
| <b>Implantación y configuración</b>  | Entorno virtualizado con GNS3 (20 h × 40 €/h)   | <b>706 €</b>       |
| <b>Integración de herramientas</b>   | Nagios (30 h) → 1 058 €<br>Syslog (15 h) → 529 €  | <b>1 587 €</b>     |
| <b>Modelos y reportes (Power BI)</b> | Modelado analítico (40 h) → 1 411 €<br>Paneles interactivos (30 h) → 1 058 €  | <b>2 469 €</b>     |
| <b>Pruebas integrales</b>            | 25 h → 882 €  | <b>882 €</b>       |
| <b>Capacitación</b>                  | 3 sesiones × 4 h → 423 €  | <b>423 €</b>       |
| <b>Documentación técnica</b>         | 20 h → 706 €  | <b>706 €</b>       |

## 2.5. Análisis de riesgos

Llevar a cabo un análisis de riesgos en nuestro proyecto es clave para **gestionar de manera adecuada los posibles retos** que puedan presentarse en la empresa. Este método nos ayudará a **determinar acciones preventivas y a minimizar las amenazas** identificadas. Durante este proceso, examinaremos detenidamente cada riesgo y, según su probabilidad de ocurrencia, los clasificaremos en tres niveles: alto, medio o bajo. Para sistematizar estos datos, utilizaremos la siguiente tabla.

| Riesgo  | Categoría        | P | I | Nivel | Estrategia de mitigación  | Responsable             |
|---|------------------|---|---|-------|---|-------------------------|
| Retraso en la obtención de licencias (Microsoft 365, Teams, etc.)   | Calendario       | 3 | 4 | 12    | Iniciar compra con antelación; definir proveedor secundario                   | Responsable de Sistemas |
| Sobrecoste por subestimación de horas en la fase de despliegue GNS3 | Presupuesto      | 3 | 4 | 12    | Incorporar un 20 % de colchón en el presupuesto; seguimiento semanal de horas | PM                      |
| Caída del entorno virtual durante las pruebas                       | Técnico          | 2 | 5 | 10    | Copias de seguridad nocturnas; snapshots antes de cada cambio mayor           | Responsable de Sistemas |
| Falta de ancho de banda en coworking para laboratorios GNS3         | Infraestructura  | 3 | 3 | 9     | Medición previa; plan alternativo con hotspot 5G dedicado                     | Responsable de Redes    |
| Vulnerabilidades en equipos demo expuestos a Internet               | Seguridad        | 2 | 4 | 8     | VLAN aislada; cortafuegos con reglas mínimas; escaneos periódicos             | Responsable de Redes    |
| Cambios de requisitos del cliente (alcance)                         | Alcance          | 2 | 4 | 8     | Acta de cambios; aprobación formal; backlog separado de "nice-to-have"        | PM + Cliente            |
| Incompatibilidad entre Nagios y la versión de Syslog elegida        | Integración      | 2 | 3 | 6     | Prototipo temprano; pruebas de compatibilidad en staging                      | Responsable de Sistemas |
| Ausencia temporal de alguno de los socios (enfermedad/viaje)        | Recursos humanos | 2 | 3 | 6     | Documentación exhaustiva; pair-review semanal; calendar interno               | PM                      |
| Incremento de costes de coworking o servicios cloud                 | Financiero       | 2 | 2 | 4     | Contrato anual fijo; opciones de cancelación flexible                         | Administración          |

Imagen 2 Análisis de Riesgos

Para cada riesgo identificado en el TFG se llevará a cabo la siguiente valoración:

1. **Probabilidad (P)**: Se asignará un valor de 1 a 3 según la probabilidad de que el riesgo se materialice (1 = baja, 2 = media, 3 = alta).
2. **Impacto (I)**: Se calificará de 1 a 5 en función de la gravedad de sus consecuencias sobre el desarrollo, la calidad o la entrega del proyecto (1 = mínimo, 5 = crítico).
3. **Nivel de Riesgo**: Se multiplican Probabilidad e Impacto ( $\text{Nivel} = P \times I$ ), obteniendo un rango entre 1 y 15 que permite ordenar los riesgos según su prioridad de atención.

A partir del valor obtenido, cada riesgo se clasifica de la siguiente manera:

- **Alto**: Nivel  $\geq 12$
- **Medio**: Nivel entre 6 y 11
- **Bajo**: Nivel  $\leq 5$

## **FASE 3: Ejecución del Proyecto**

Durante esta fase, se utilizarán los recursos previamente identificados y estimados para garantizar que el desarrollo se ajuste a lo planeado.

### **3.1. Descripción de las tareas realizadas**

Implementación de la topología en GNS3

Despliegue de la GNS3 VM sobre VMware Workstation/Player en modo de virtualización anidada.

Creación de routers Cisco ISR, switch Catalyst virtual y PCs ligeros (VPCS).

Integración de servicios de monitorización y registro

Nagios y Syslog, conectados a la red virtual.

Verificación de recolección de métricas y envío de logs.

Exportación y visualización de datos en Power BI

Recopilación de información (outputs de CLI y logs).

Modelado de datos en Power BI y diseño de dashboards con KPIs clave.

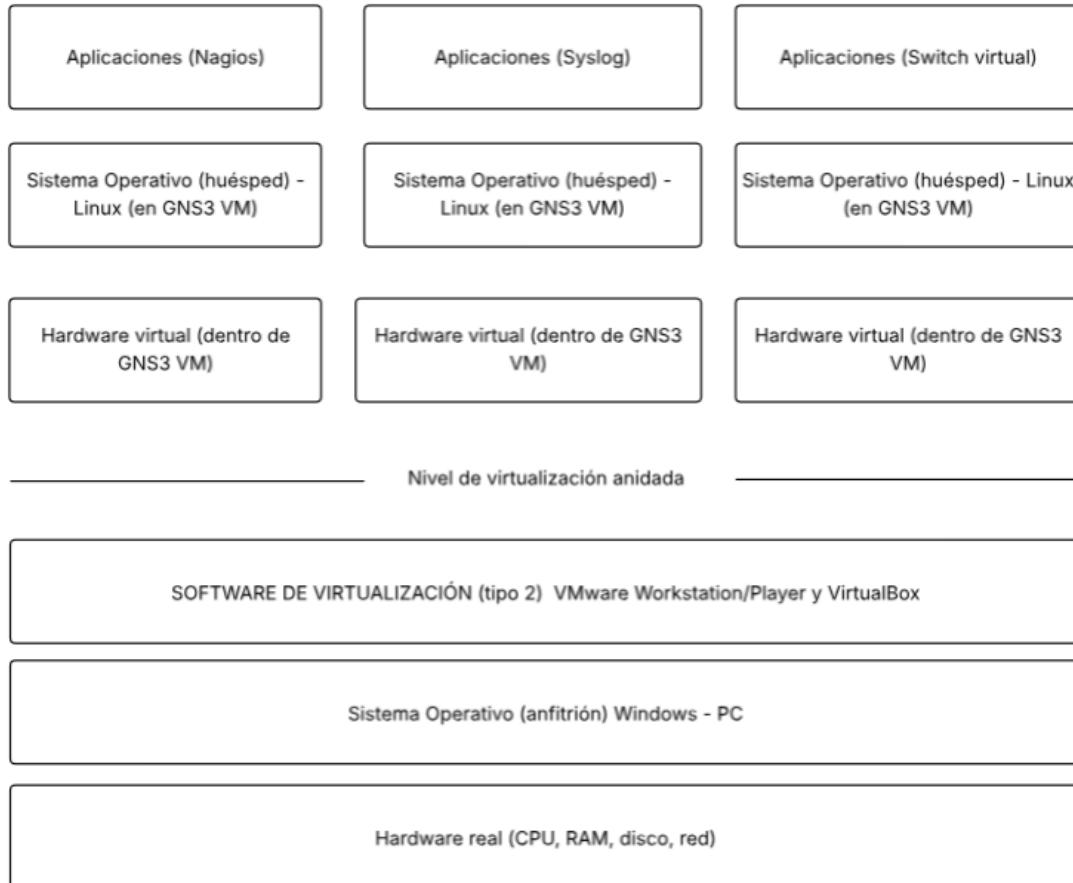
Informe de actividades y conclusiones

Elaboración de un informe intermedio con capturas de pantalla.

Análisis de resultados, logros y lecciones aprendidas.

### **3.2. Arquitectura de máquinas virtuales**

Para desplegar nuestra topología en un entorno controlado y reproducible, configuramos un nivel de virtualización anidada sobre VMware Workstation/Player y VirtualBox, ejecutando en su interior la GNS3 VM que alberga los dispositivos virtuales.



*Imagen 3 Arquitectura de la virtualización*

Estos niveles garantizan que los servicios de monitorización y logging convivan con emulaciones de routers, switchs, servidores y PCs ligeros, sin comprometer la fidelidad de la simulación.

### 3.2.2 Elección del hipervisor

Para las VMs de servidor y la propia GNS3 VM, optamos por **VMware Workstation/Player** y **Virtualbox** (hipervisores de tipo 2), que ofrecen:

- **Facilidad de despliegue:** interfaz gráfica intuitiva y rápida creación de VMs.
- **Compatibilidad:** amplio soporte a sistemas operativos huéspedes (Linux, Windows).
- **Flexibilidad:** snapshots y clonación para agilizar pruebas y restauración.
- **Rendimiento moderado:** idóneo en laboratorios, con un impacto aceptable en entornos de sobremesa.

### 3.2.3 Requisitos de hardware recomendados

Para obtener un rendimiento fluido en simulaciones de red con múltiples dispositivos, se aconseja:

| Componente            | Mínimo                                      | Recomendado                                |
|-----------------------|---|--|
| <b>CPU</b>            | Quad-core ×2.5 GHz                          | Hexa-core o más, con VT-x/AMD-V habilitado |
| <b>RAM</b>            | 16 GB                                       | 32 GB o superior                           |
| <b>Almacenamiento</b> | SSD 256 GB                                  | SSD NVMe ≥ 512 GB                          |
| <b>Red</b>            | Gigabit Ethernet                            | Gigabit o superior                         |
| <b>SO anfitrión</b>   | Windows 10/11 Pro o Linux moderno (64 bits) | —  |

### 3.2.4 GNS3 VM: descripción y ventajas

La **GNS3 VM** es una máquina virtual que actúa como “back-end” para GNS3, trasladando la carga de emulación a un entorno Linux optimizado:

- **Sistema base:** Ubuntu LTS con QEMU, Dynamips y Docker preinstalados.
- **Arquitectura ligera:** sacrifica interfaz gráfica en favor de un uso eficiente de CPU y RAM.
- **Integración:** conexión directa con el cliente GNS3 a través de socket para desplegar routers, switchs e interfaces con un clic.
- **Escalabilidad:** permite levantar imágenes de virtualización de red más pesadas (vMX, vSRX, FortiGate) con menor sobrecarga que en el host.

De este modo, se separa la capa de simulación (GNS3 VM) de la del hypervisor principal, mejorando la estabilidad y la reproducibilidad de las topologías complejas.

### 3.3. Introducción general

Este proyecto propone una topología de red virtualizada, creada con GNS3 y VirtualBox, que combina routers Cisco ISR, un switch de capa 2 y PCs virtuales (VPCS). Su objetivo es demostrar la integración de servicios de monitorización, registro de eventos y protocolos de enrutamiento en un entorno controlado, siguiendo los principios del modelo OSI.

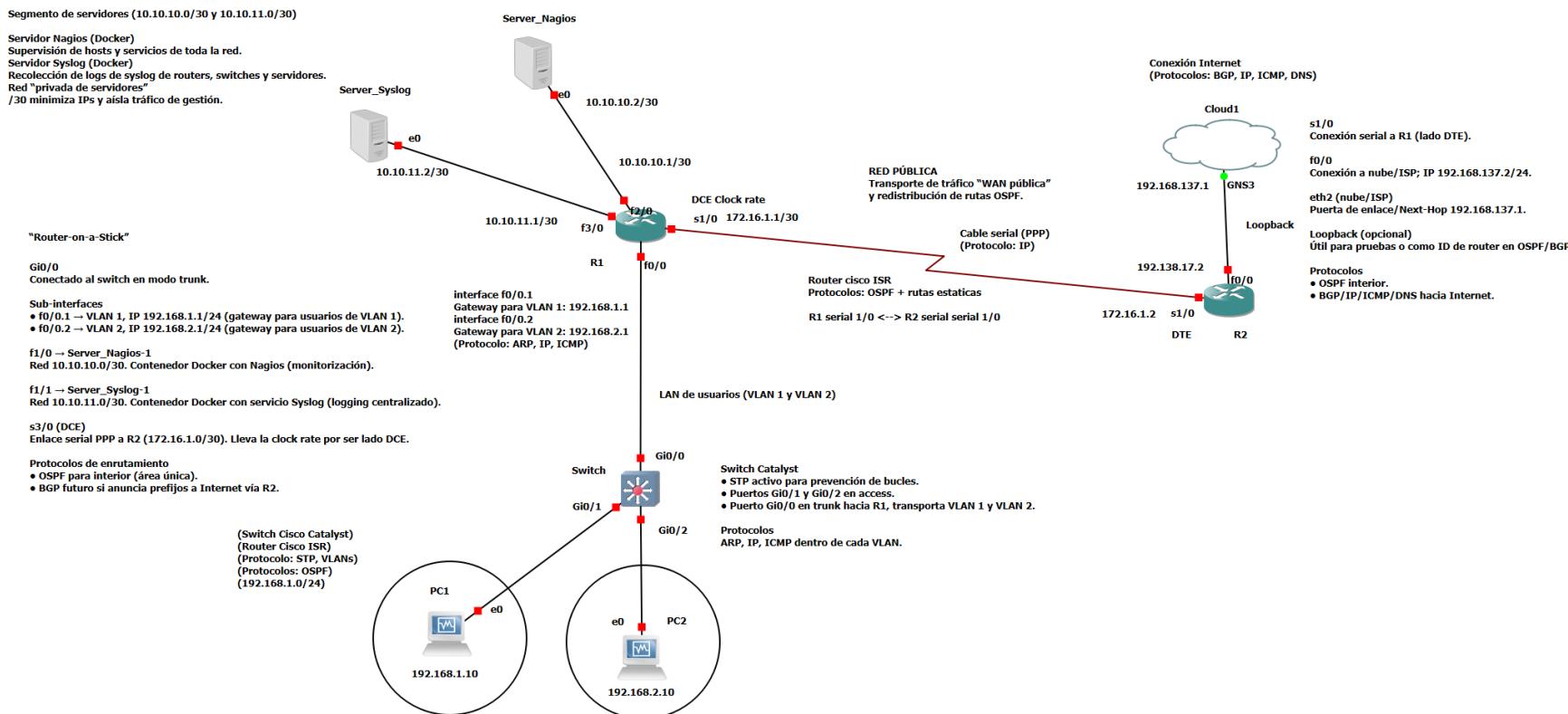


Imagen 4 Prototipo de la virtualización en Gns3

### 3.3.1 Componentes principales

#### 1. Servidores (Capa 7/1)

- **Nagios:** Herramienta de monitorización de red aislada que facilita su despliegue, actualización y gestión.
- **Syslog:** Servicio de recogida y almacenamiento de mensajes de registro, ejecutado independiente para asegurar la separación de funciones.

#### 2. Routers Cisco ISR (Capa 3)

- **R1:** Encaminamiento entre los segmentos privados de servidores, la red local de usuarios y el enlace público.
- **R2:** Enlace hacia la “nube” de Internet simulada, gestionando protocolos OSPF y BGP para reflejar entornos reales.

#### 3. Switch Cisco Catalyst Virtual (Capa 2)

- Segmenta el tráfico en **VLAN 1** y **VLAN 2**, aplicando STP para evitar bucles y proporcionando acceso de nivel de enlace a los PCs virtuales.

#### 4. Enlace serial PPP (Capa 2)

- Simula un enlace punto a punto entre R1 y R2 para transporte de paquetes IP.

#### 5. Nube de Internet (Capa 3–7)

- Representa la conectividad externa, incluyendo resolución DNS y protocolos de control (ICMP, BGP).

#### 6. PCs virtuales (VPCS) (Capa 3/4)

- Dos estaciones de usuario (Ubuntu y Windows), cada una en su VLAN respectiva, para pruebas de conectividad, rendimiento y monitorización.

### 3.3.2 Interacción según el modelo OSI

- **Capa 7 (Aplicación):** Generación de tráfico por Nagios, Syslog y aplicaciones en los PCs.
- **Capa 4 (Transporte):** Encapsulación TCP/UDP para servicios de monitorización y registro.
- **Capa 3 (Red):** Enrutamiento IP dinámico con OSPF y BGP, segmentación de subredes privadas y públicas.
- **Capa 2 (Enlace):** VLANs y PPP para segmentar dominios de difusión y simular enlaces dedicados.

- **Capa 1 (Física virtualizada):** Conexiones por interfaces virtuales en GNS3, que emulan cables Ethernet y seriales.

### 3.4 Instalación de los requisitos previos

En este apartado se detallan todos los programas necesarios para llevar a cabo la simulación, incluyendo VirtualBox, GNS3 con su respectivo servidor VM, imágenes de Cisco IOS, sistemas operativos Ubuntu y, opcionalmente, Docker.

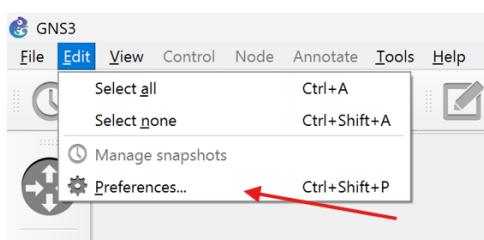
Antes de iniciar, instala los siguientes programas:

- **VirtualBox:** Software de virtualización para máquinas virtuales (Nagios, Syslog, PC).
- **GNS3:** Plataforma de simulación y emulación de redes.
  - Descarga desde <https://www.gns3.com>.
  - Descarga e instala también la **GNS3 VM Server** desde la página oficial de GNS3 (<https://www.gns3.com>).
  - Asegúrate que la versión del **GNS3 VM Server** coincide con la versión de la aplicación GNS3 instalada. Por ejemplo, en este proyecto se utiliza la versión **2.2.53**.
  - Esto permite que GNS3 gestione los dispositivos Cisco (routers, switches) y la Cloud directamente desde el servidor GNS3, optimizando el rendimiento.
- **Docker** (opcional): Para servidores en contenedores.
- **Imagen IOS Cisco:** Ejemplo, c7200 o c3745 para routers en GNS3.
- **Ubuntu Server 22.04 LTS (64-bit):** Sistema operativo para servidores.
- **Ubuntu Desktop 22.04 (64-bit):** Sistema operativo para PCs usuarios.

### 3.5 Configuración de GNS3 para uso con GNS3 VM Server

Se siguen estos pasos detallados para configurar GNS3 utilizando el servidor local y el GNS3 VM:

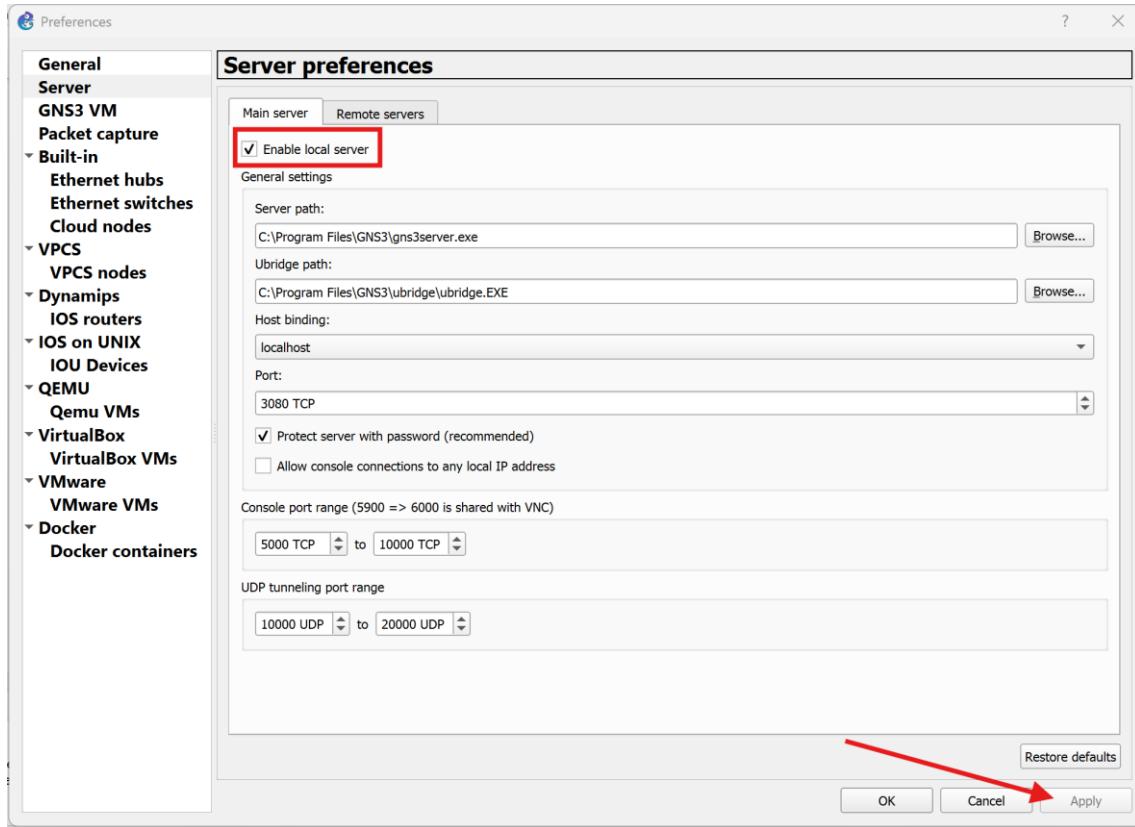
1. Abre GNS3 → Menú Editar → Preferencias:



2. Configuración del Servidor local:

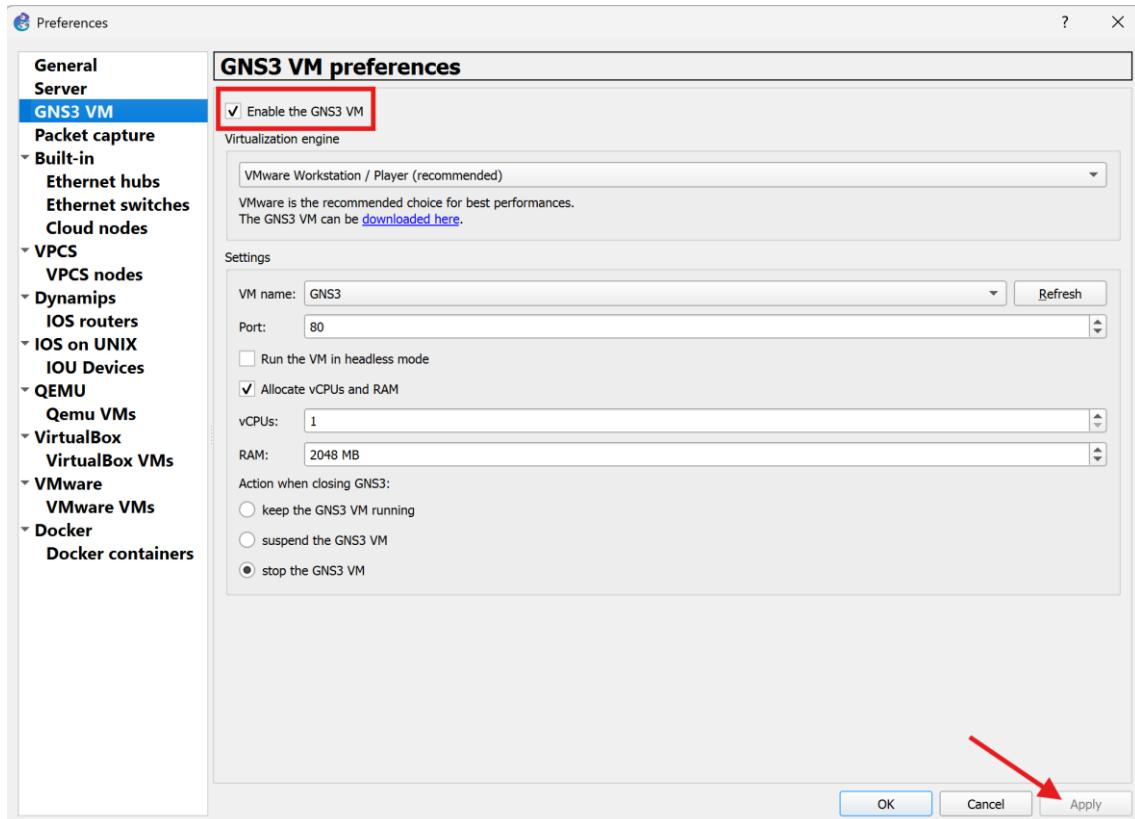
- Selecciona la opción **Server** del panel izquierdo.
- Asegúrate de marcar **Enable local server**.
- Verifica la ruta correcta del servidor:
  - **Server path:** Ejemplo C:\Program Files\GNS3\gns3server.exe
  - **Ubridge path:** Ejemplo C:\Program Files\GNS3\ubridge\ubridge.EXE

- **Host binding:** Debe estar configurado como localhost.
- **Puerto:** Por defecto 3080 TCP.



### 3. Configuración del GNS3 VM:

- Selecciona la opción **GNS3 VM** del panel izquierdo.
- Marca la casilla **Enable the GNS3 VM**.
- Selecciona **VMware Workstation / Player (recommended)** como motor de virtualización.
- Define los recursos para la VM:
  - **vCPUs:** 1
  - **RAM:** 2048 MB (o más, según recursos disponibles)
- Selecciona la opción **Stop the GNS3 VM** al cerrar GNS3 para optimizar recursos.

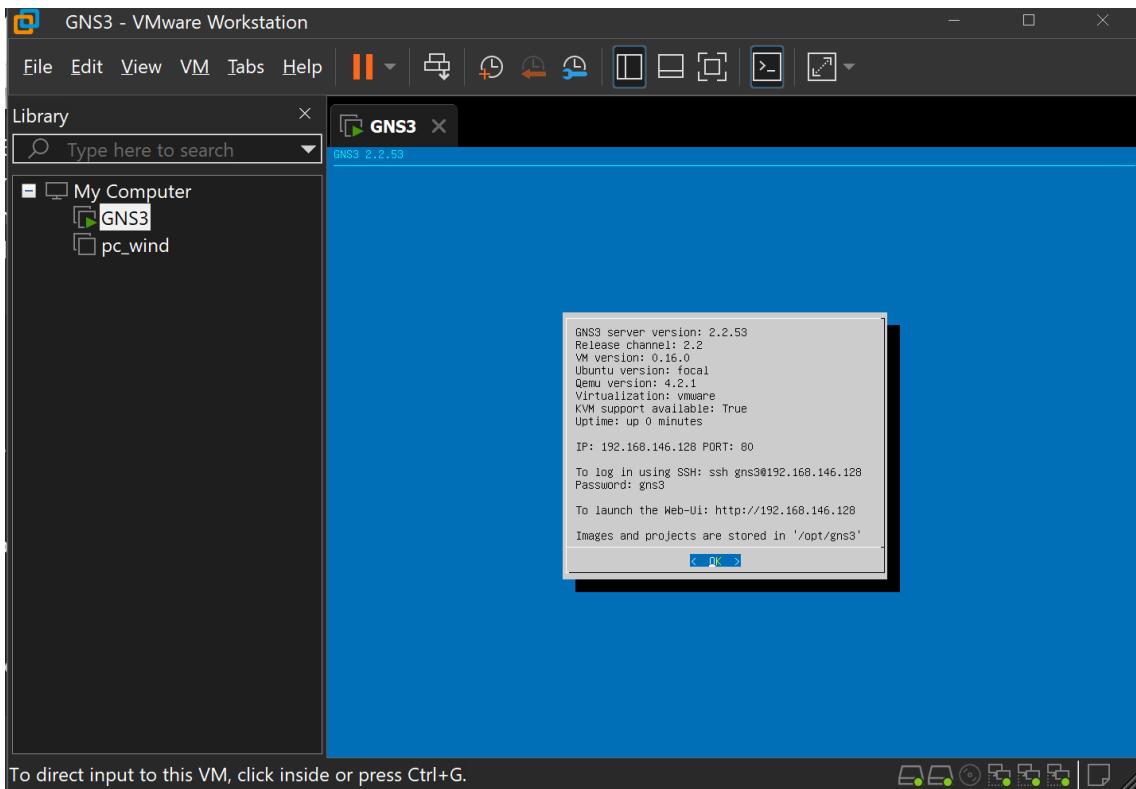


Aplica los cambios y confirma con **OK**.

Una vez configurada la máquina virtual GNS3 VM en VMware Workstation, es fundamental verificar que la conexión entre esta máquina virtual y la aplicación GNS3 se haya establecido correctamente.

Cuando el finalice el arranque completo en VMware, en la ventana emergente dentro de la máquina virtual GNS3 VM debe proporcionar información importante, como:

- **Versión del servidor GNS3 VM** (en este caso, la versión utilizada es **2.2.53**).
- Estado de virtualización y soporte KVM (debe indicar KVM support available: True).
- Dirección IP asignada automáticamente mediante DHCP (por ejemplo: 192.168.146.128) y el puerto (80 por defecto).

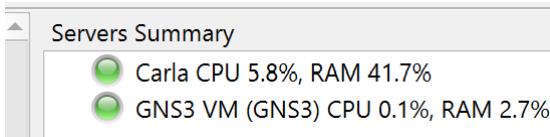


*Imagen 5 Virtualización de GNS3 VM*

Para comprobar el correcto funcionamiento en la aplicación GNS3, en el panel izquierdo inferior llamado Servers Summary, se verifica el estado de los servidores configurados.

Se observa al menos dos servidores activos con indicadores de estado en color verde:

- **Servidor local** (identificado como el nombre de tu ordenador, en mi caso, Carla): indica el uso de CPU y RAM reales.
- **Servidor GNS3 VM**: también muestra la carga actual en términos de CPU y RAM utilizados por esta máquina virtual.



Interpretación de indicadores:

- **Color Verde**: Significa que la conexión entre la aplicación GNS3 y los servidores está correctamente establecida, y ambos servidores (local y VM) están operativos.
- Cualquier indicador en color rojo o gris implica un problema de conectividad o configuración que debe resolverse antes de avanzar.

### 3.5.1 Funcionamiento y distribución de carga en GNS3

GNS3 maneja de manera eficiente la carga de trabajo entre dos servidores principales:

- **Servidor Local (Computadora real):**
  - Utiliza directamente los recursos físicos de la computadora anfitriona (CPU, RAM).
  - Normalmente maneja dispositivos ligeros como VMs externas gestionadas por VirtualBox, QEMU, Docker o nodos menos demandantes.
- **GNS3 VM Server:**
  - Máquina virtual dedicada, diseñada específicamente para GNS3.
  - Ideal para gestionar dispositivos Cisco (routers IOS, switches avanzados) y nodos que requieren una virtualización más intensiva.
  - Permite un mejor rendimiento y distribución de recursos, aliviando la carga sobre el servidor local.

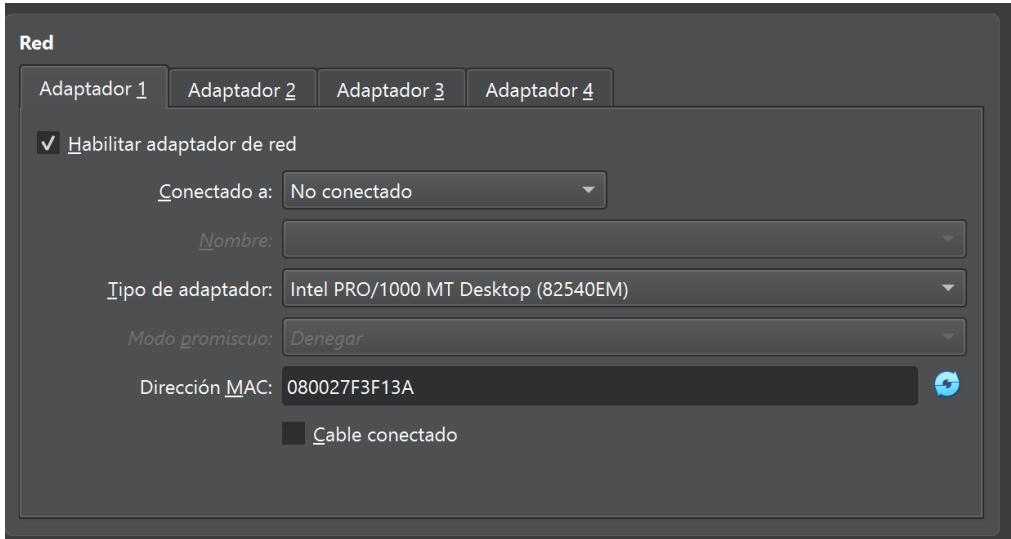
Durante el diseño y configuración de la topología, GNS3 permite seleccionar el servidor sobre el cual se ejecutará cada dispositivo, permitiendo equilibrar la carga según las necesidades específicas de rendimiento y optimización.

### 3.6 Configuración detallada de VirtualBox

Aquí se describen los pasos específicos para preparar las máquinas virtuales necesarias (servidores y PCs usuarios), asignando recursos adecuados (RAM, disco, red) y configurando cada máquina virtual para su correcto funcionamiento dentro de la topología.

#### 3.6.1 Máquina virtual para Servidor Nagios

1. Se abre **VirtualBox → Nueva:**
  - Nombre: **ServidorNagios**
  - Tipo: **Linux**
  - Versión: **Ubuntu (64-bit)**
2. RAM: **2048 MB**
3. Disco: **20 GB VDI (Virtual Disk Image)**
4. Configuración de Red:
  - Adaptador 1: **No conectado** (habilitar cable más tarde en GNS3)



### Instalación en consola del servidor:

```
sudo apt update && sudo apt upgrade -y  
sudo apt install docker.io docker-compose -y  
sudo usermod -aG docker $USER  
sudo reboot
```

### Opcional (Docker):

```
docker run -d --name nagios -p 8080:80 jasonrivers/nagios:latest
```

### 3.6.2 Máquina virtual para Servidor Syslog

1. Se clona la VM de Nagios como **Linked Clone**.
2. Nombre: **ServidorSyslog**
3. RAM: **1024 MB**

### Instalación básica:

```
sudo apt update  
sudo apt install rsyslog rsyslog-relp -y
```

### Opcional (Docker):

```
docker run -d --name syslog -p 514:514/udp -p 514:514 adubkov/Rsyslog
```

### 3.6.3 Máquinas virtuales para usuarios (PC1 y PC2)

1. Se crea nuevas máquinas virtuales:

- o Nombres: **PC1, PC2**
- o Sistema Operativo: **Ubuntu Desktop 22.04**
- o RAM: **1024 MB**
- o Disco: **15 GB**

### 3.7 Configuración de red con IP estática

Se edita Netplan en cada VM para asignar IP estática.

Ejemplo configuración **ServidorNagios**:

```
sudo nano /etc/netplan/01-netcfg.yaml
```

Contenido:

network:

version: 2

renderer: networkd

ethernets:

ens3:

dhcp4: no

addresses: [192.168.56.10/24]

gateway4: 192.168.56.1

nameservers:

addresses: [8.8.8.8, 8.8.4.4]

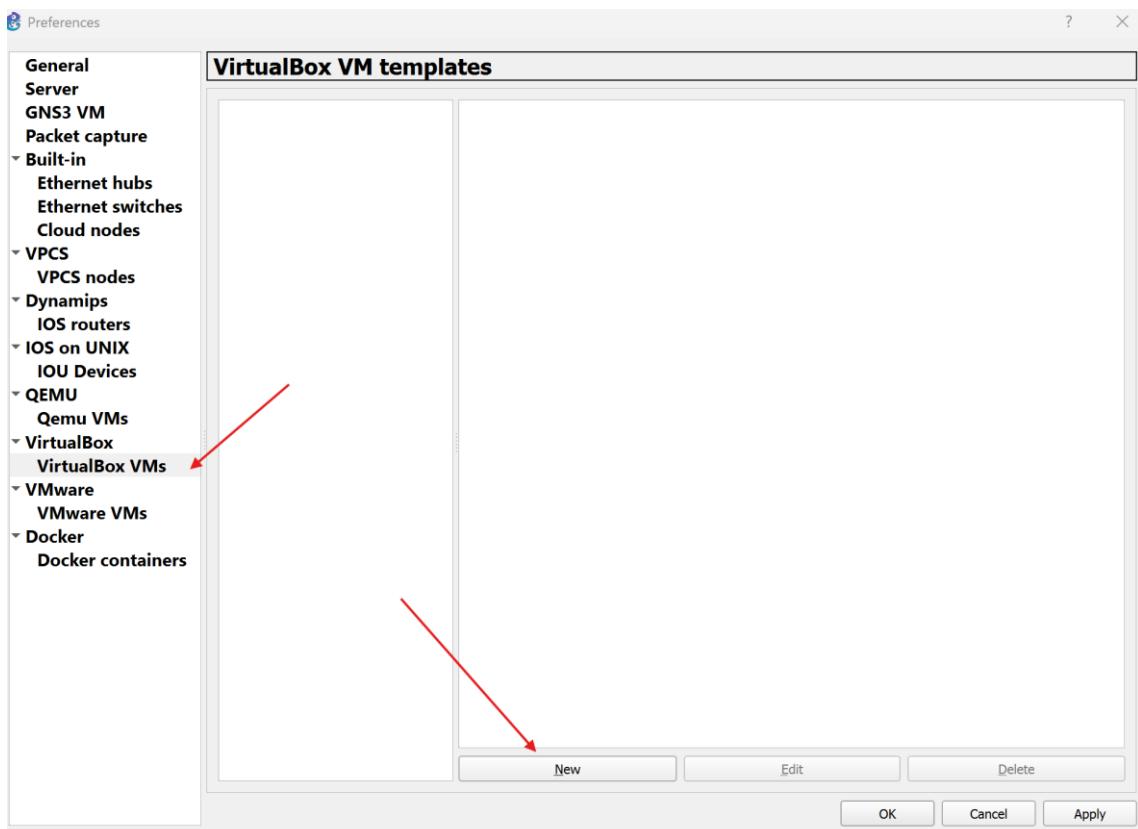
Se aplica la configuración:

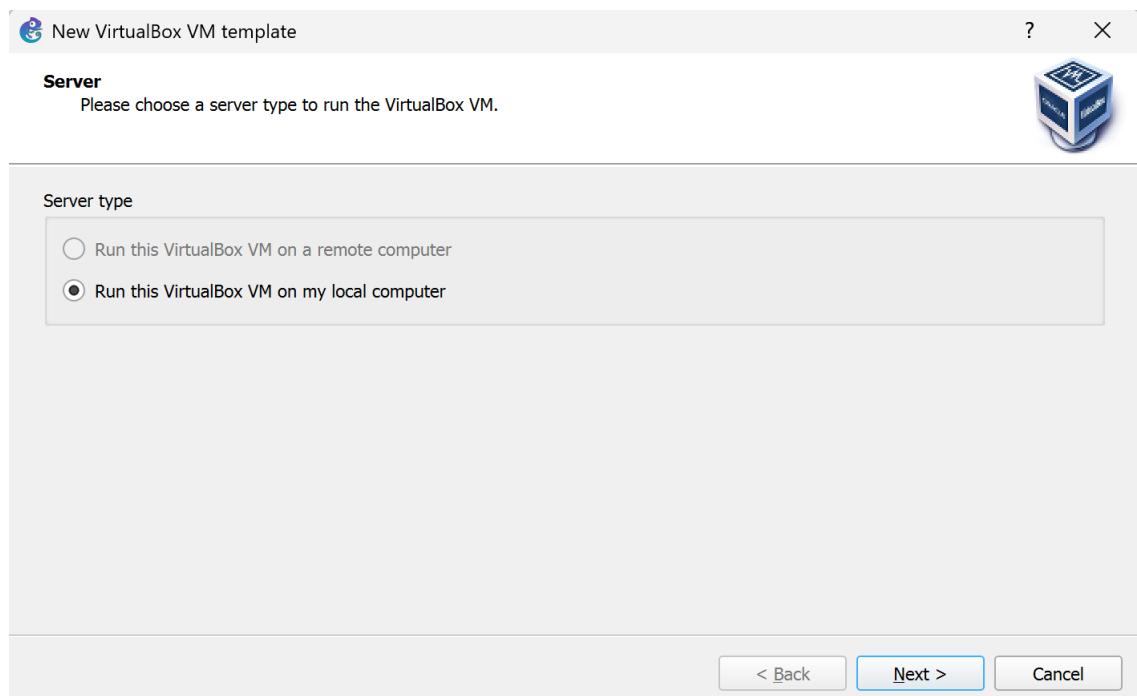
```
sudo netplan apply
```

| Máquina        | Interfaz | Dirección IP      | Gateway      |
|----------------|----------|-------------------|--------------|
| ServidorNagios | ens3     | 192.168.56.10/24  | 192.168.56.1 |
| ServidorSyslog | ens3     | 192.168.56.20/24  | 192.168.56.1 |
| PC1            | ens3     | 192.168.56.101/24 | 192.168.56.1 |
| PC2            | ens3     | 192.168.56.102/24 | 192.168.56.1 |

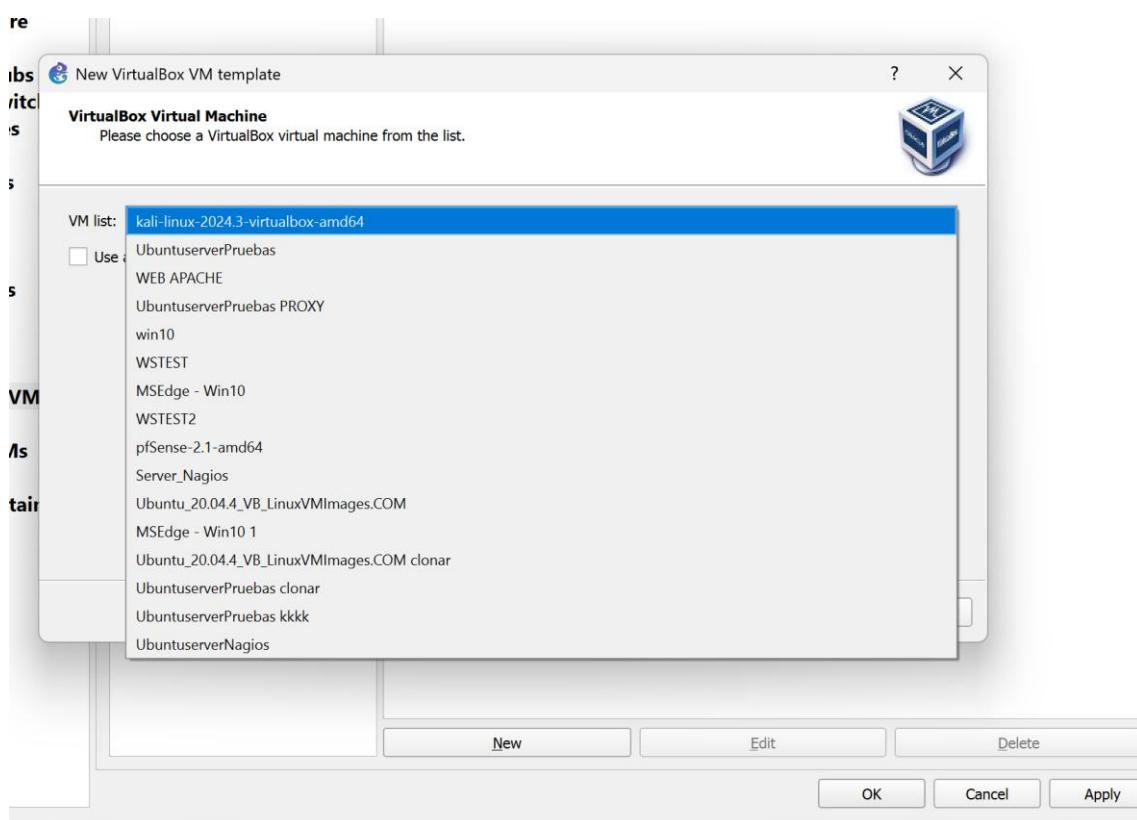
### 3.8 Añadir Las VMs a GNS3

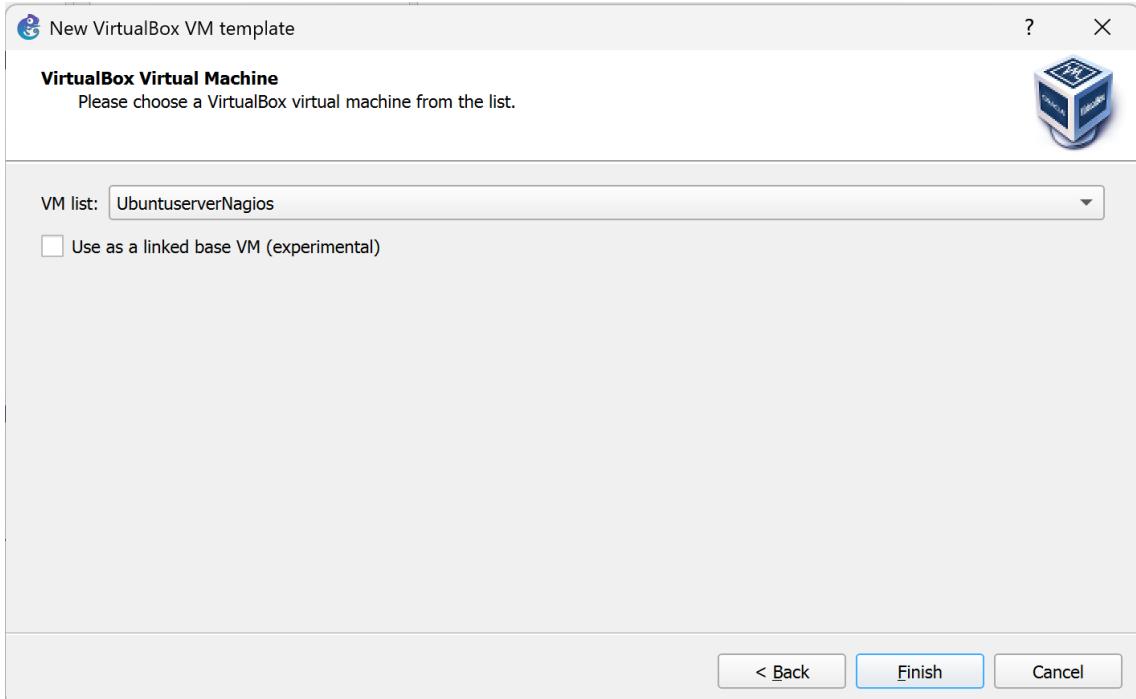
1. **Se abre GNS3 → Nuevo proyecto → TFG**
2. **Editar → Preferencias**
3. **VirtualBox VMs → New**
  - Se selecciona ServidorNagios, marca *Enable console support* → **Finish**



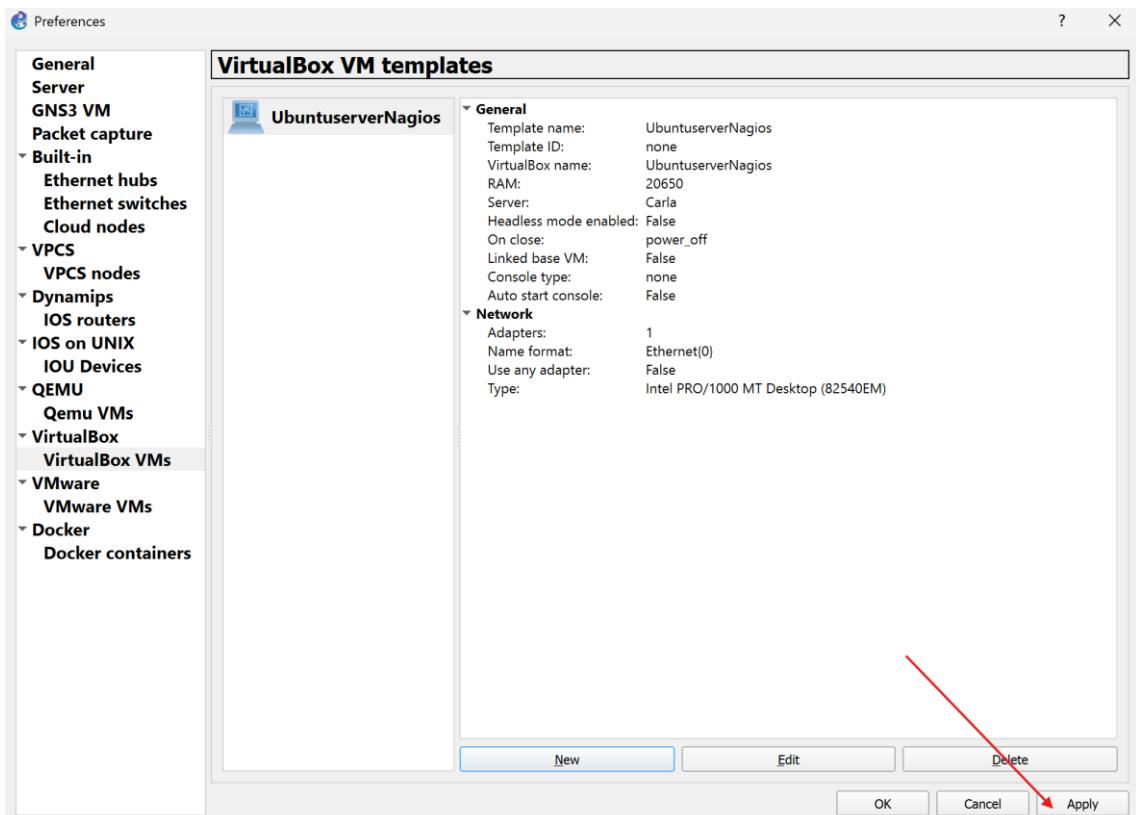


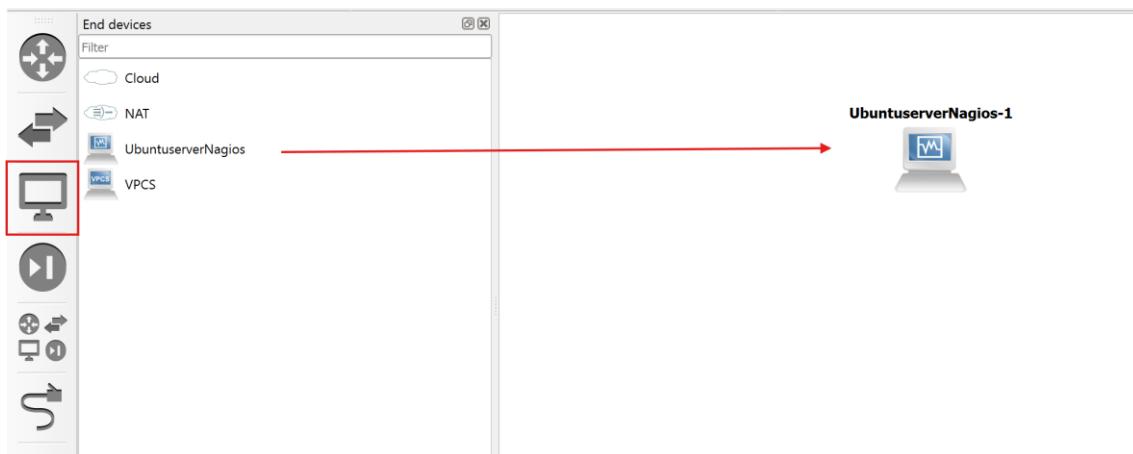
Se busca la maquina indicada.





Cuando se termina, se aplica a la topología,





- Repite para ServidorSyslog, PC1 y PC2.

GNS3 solo crea un *wrapper*; la VM real sigue gestionada por VirtualBox.

### 3.9 Configurar GNS3 y añadir los dispositivos

Se detalla la inclusión de dispositivos de red dentro de GNS3, como routers Cisco y switch virtual, junto con la asignación de imágenes IOS, configuración básica de interfaces, y protocolos de enrutamiento según las necesidades de la simulación. Corregir ;

#### 3.9.1 Añadir y configurar dispositivos en GNS3

Se agrega y configura los siguientes dispositivos:

- **Routers Cisco (R1 y R2):**
  1. Abrir la sección de plantillas
  - Se abre GNS3.
  - En la barra izquierda, se hace clic en el ícono de engranaje o ve a **Editar** → **Preferencias**.
  - Se navega a **Dynamips** → **IOS routers**.

#### 2. Crear una nueva plantilla

Haz clic en New (abajo a la izquierda).

En la ventana "New template", se selecciona:

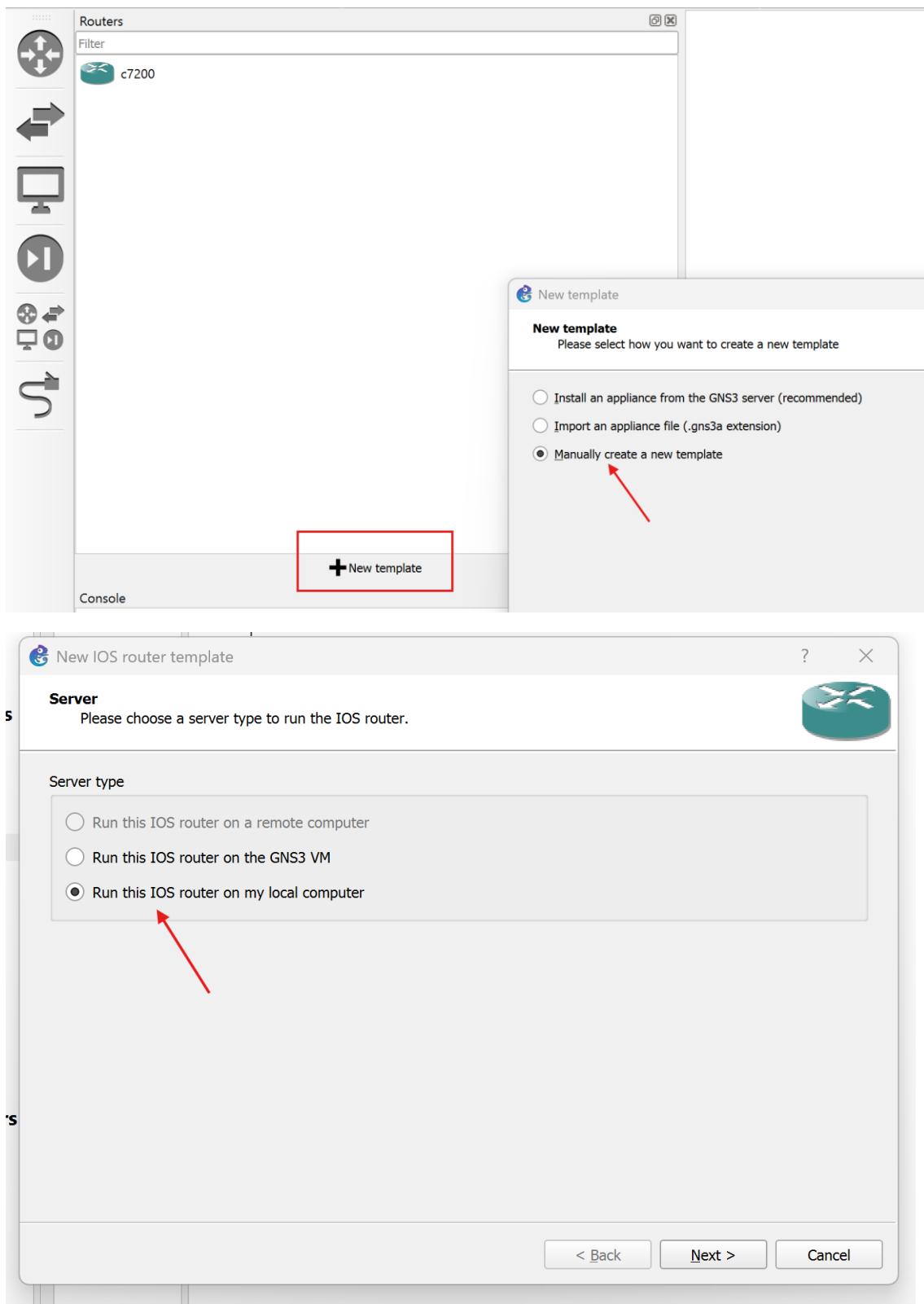
Manually create a new template.

Luego se hace clic en **Next**.

### 3. Seleccionar dónde se ejecutará el router

Elije: **Run this IOS router on my local computer.**

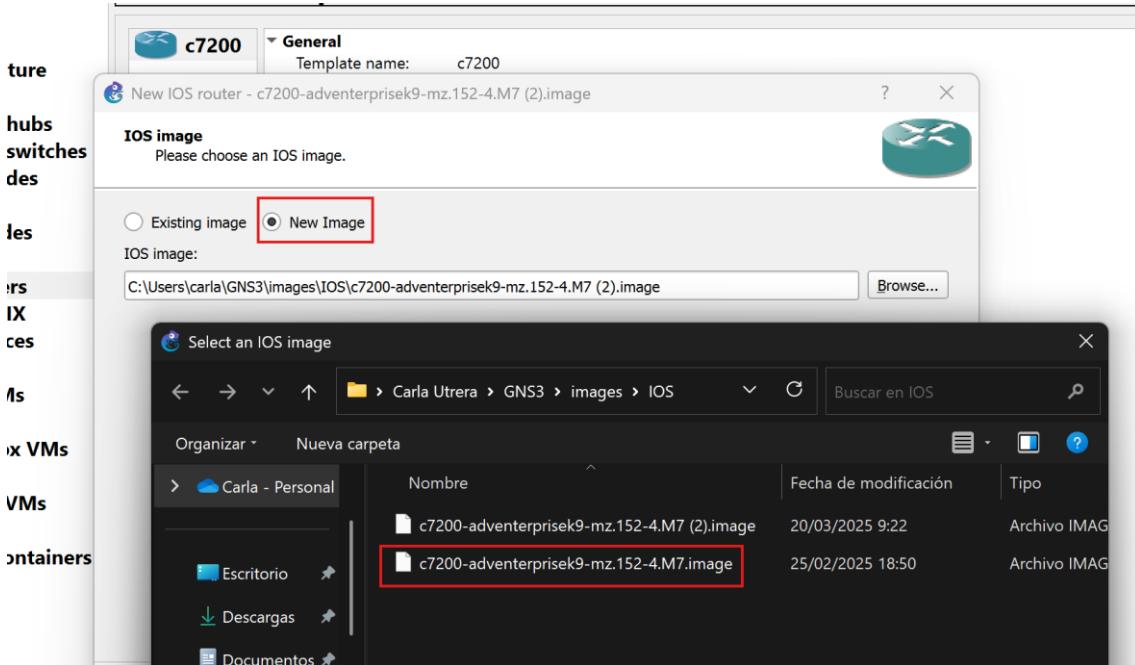
(También se puede usar la GNS3 VM si prefieres rendimiento superior).



#### 4. Cargar imagen IOS

En "Image", se selecciona el archivo. image del IOS Cisco, por ejemplo: **c7200-adventerprisek9-mz.152-4.S5.image**

GNS3 lo copiará a su repositorio interno. Dale unos segundos.



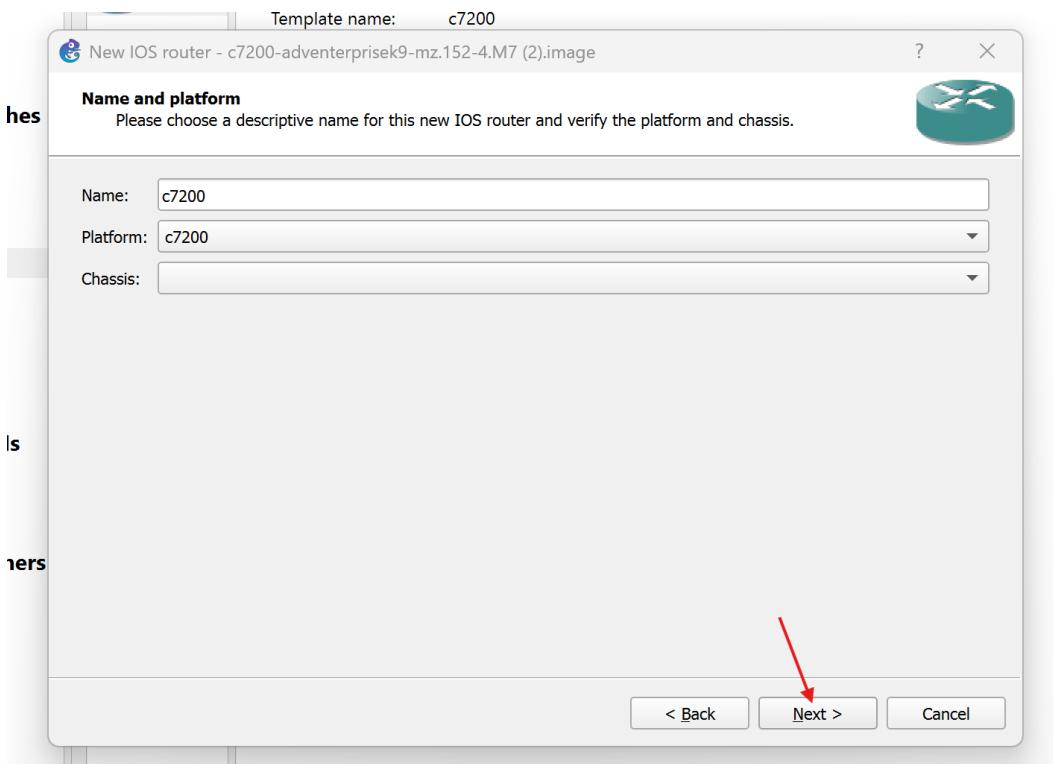
#### 5. Configurar nombre, plataforma y chasis

**Name:** Escribe un nombre descriptivo, por ejemplo: R-C7200.

**Platform:** Deja **c7200**.

**Chassis:** Elige el tipo correcto (por defecto c7200).

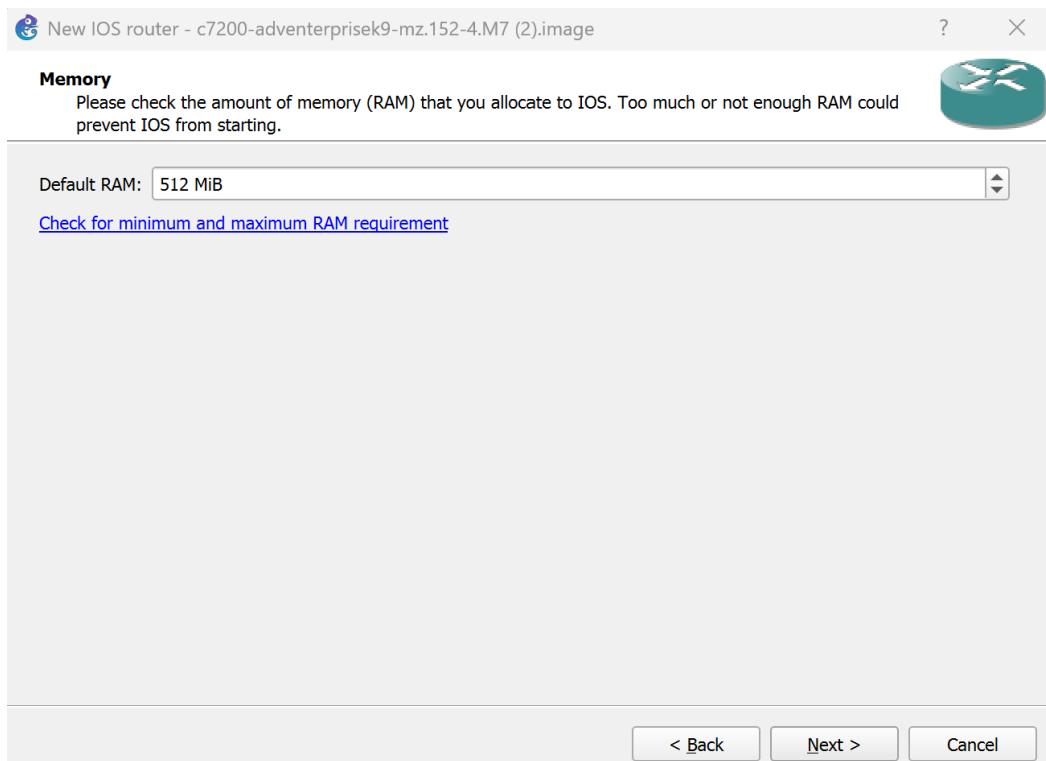
Se hace clic en **Next**.



## 6. Asignar memoria RAM

Se asigna **512 MB** de RAM (valor recomendado para c7200).

Se hace clic en **Next**.

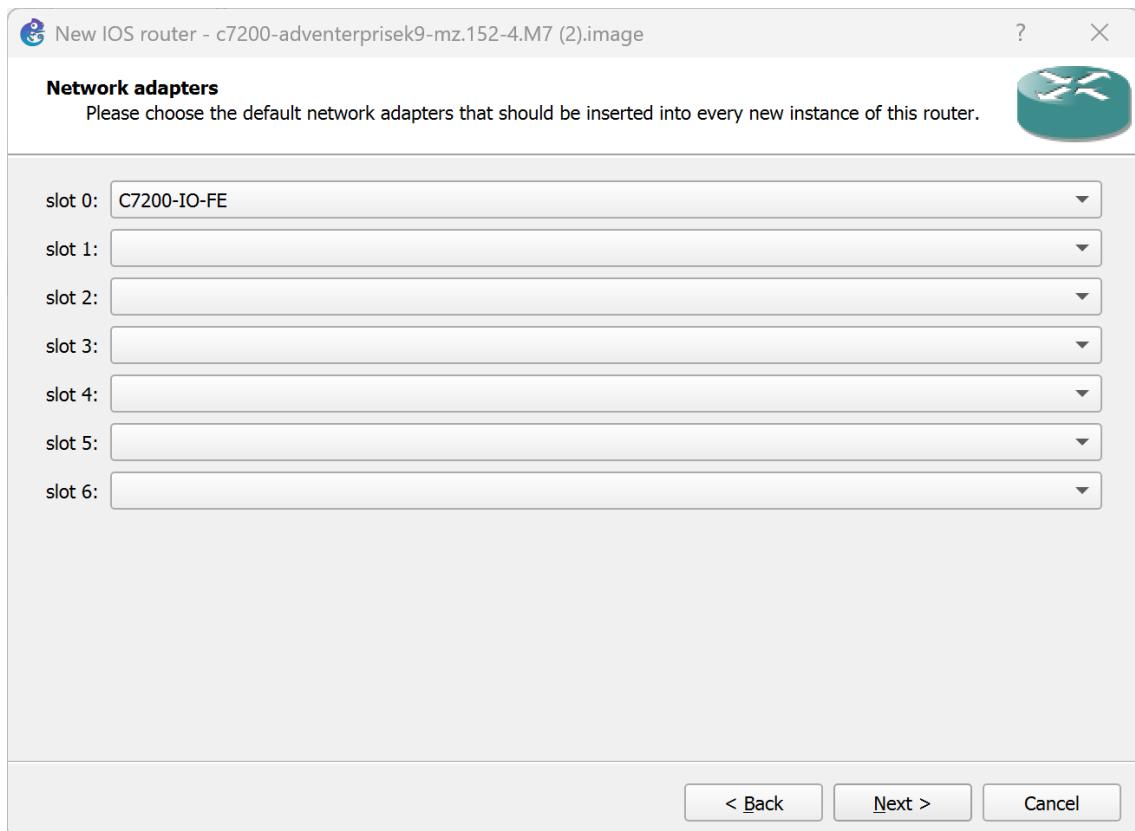


## 7. Seleccionar adaptadores de red

Slot 0: Se elige C7200-IO-FE

Slot 1: Puedes usar PA-4T+ o PA-GE (según tipo de interfaz que necesites).

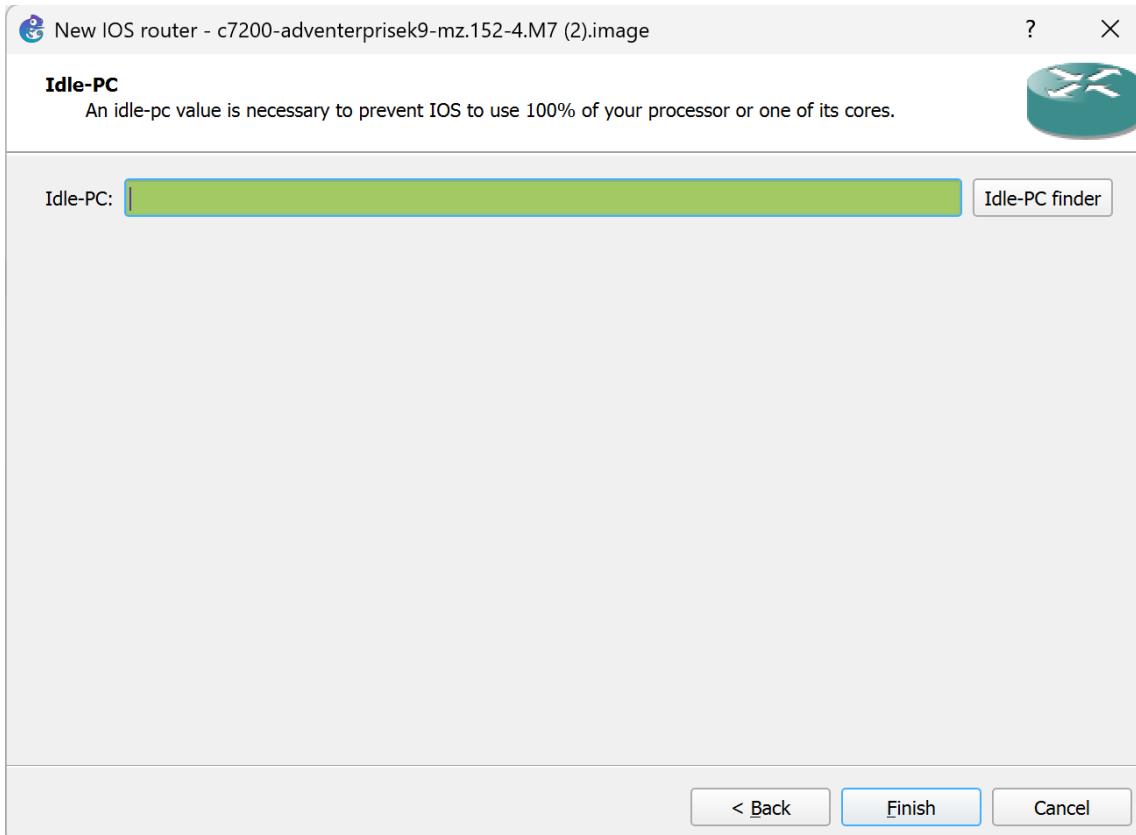
Se hace clic en **Next**.



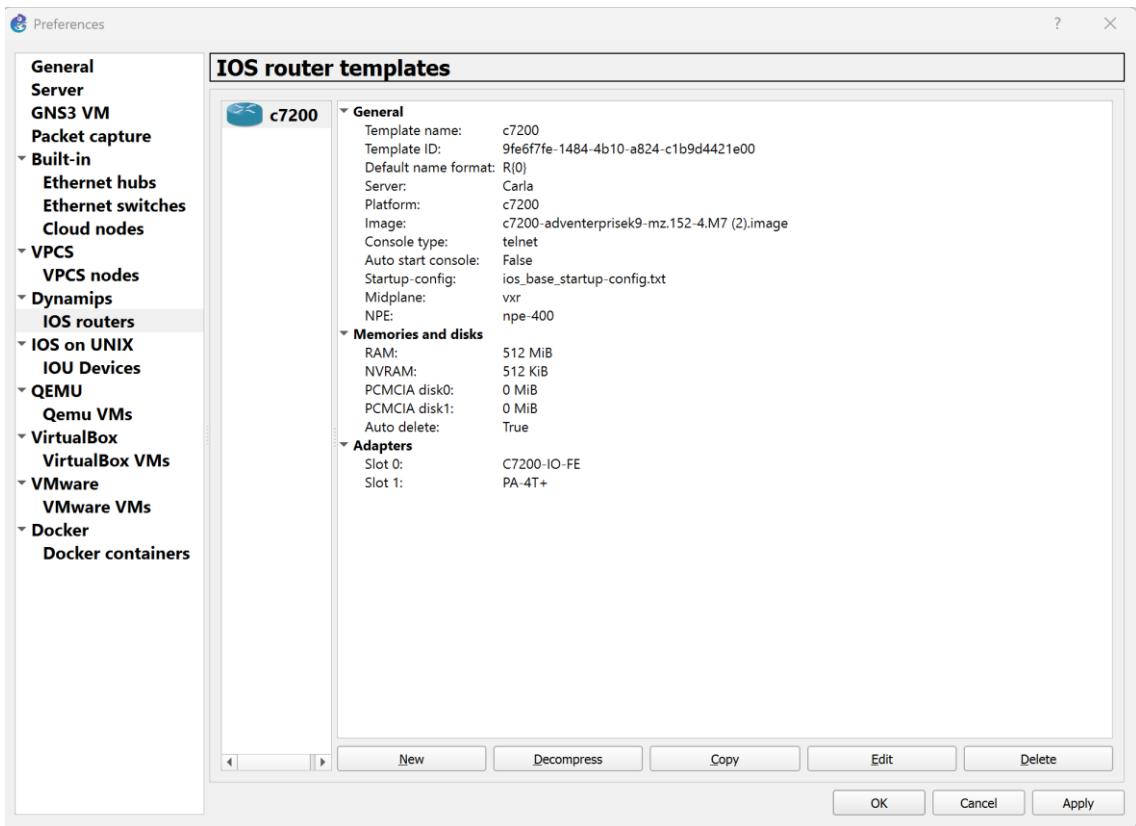
## 8. Configurar Idle-PC

Esta opción es importante para evitar que el router consuma el 100% del CPU.

Se hace clic en **Idle-PC finder** (GNS3 sugerirá un valor automáticamente).



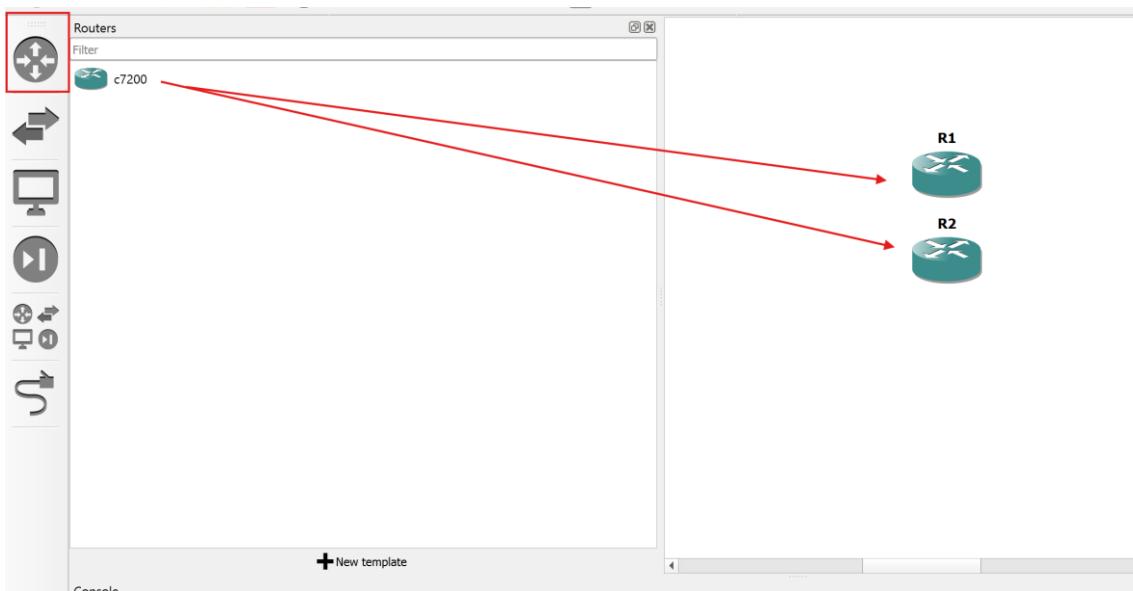
Se hace clic en **Finish**.



### 3.9.2 Añadir los routers al proyecto

En la lista de dispositivos ahora aparece el router c7200.

Se arrastra dos veces al área de trabajo para crear R1 y R2.



- **Switch Cisco Catalyst:**

### 3.9.3 Uso de un Switch Cisco Catalyst de Capa 3

En la topología de red diseñada, se tenía previsto utilizar un switch Cisco Catalyst convencional (de Capa 2). Sin embargo, el switch por defecto disponible en algunos entornos de simulación (como GNS3 o Packet Tracer) **no soporta el encapsulamiento IEEE 802.1Q**, necesario para implementar subinterfaces en un esquema **Router-on-a-Stick**, ni tampoco permite la configuración completa de VLANs con capacidades de enrutamiento.

Por esta razón, se decidió utilizar un **Switch de Capa 3 (Layer 3 Switch)**, específicamente un modelo **Cisco Catalyst multilayer**, que sí permite:

- **Encapsulamiento DOT1Q** para tráfico VLAN entre el router y el switch.
- **Soporte de trunking y subinterfaces.**
- Configuración avanzada de VLANs y asignación por puerto.
- Enrutamiento entre VLANs en caso de requerirse en futuras expansiones.

Esto garantiza una operación correcta del diseño con múltiples VLANs y una configuración realista orientada a entornos empresariales.

## 1. Se crea un nuevo template

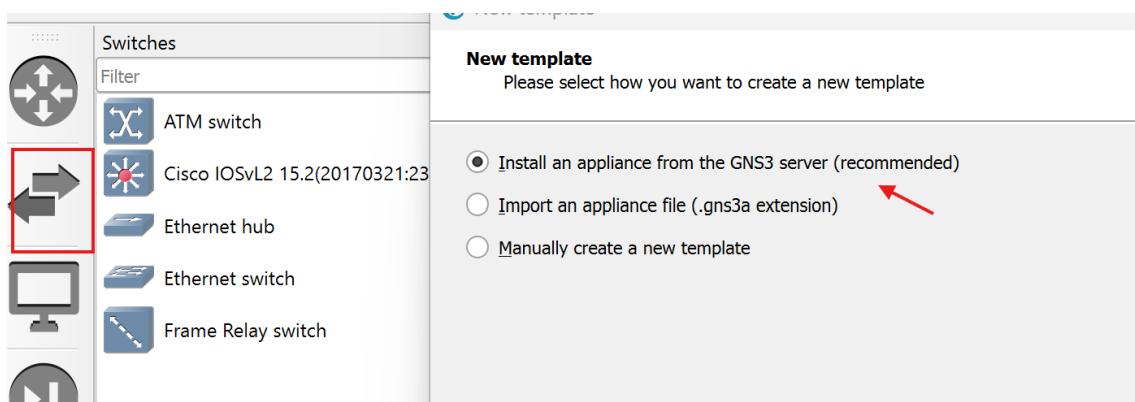
En GNS3, se hace clic en el icono de Switches en la barra lateral.

Luego, abajo, se presiona **+ New template**.

Se selecciona la opción:

Install an appliance from the GNS3 server (recommended)

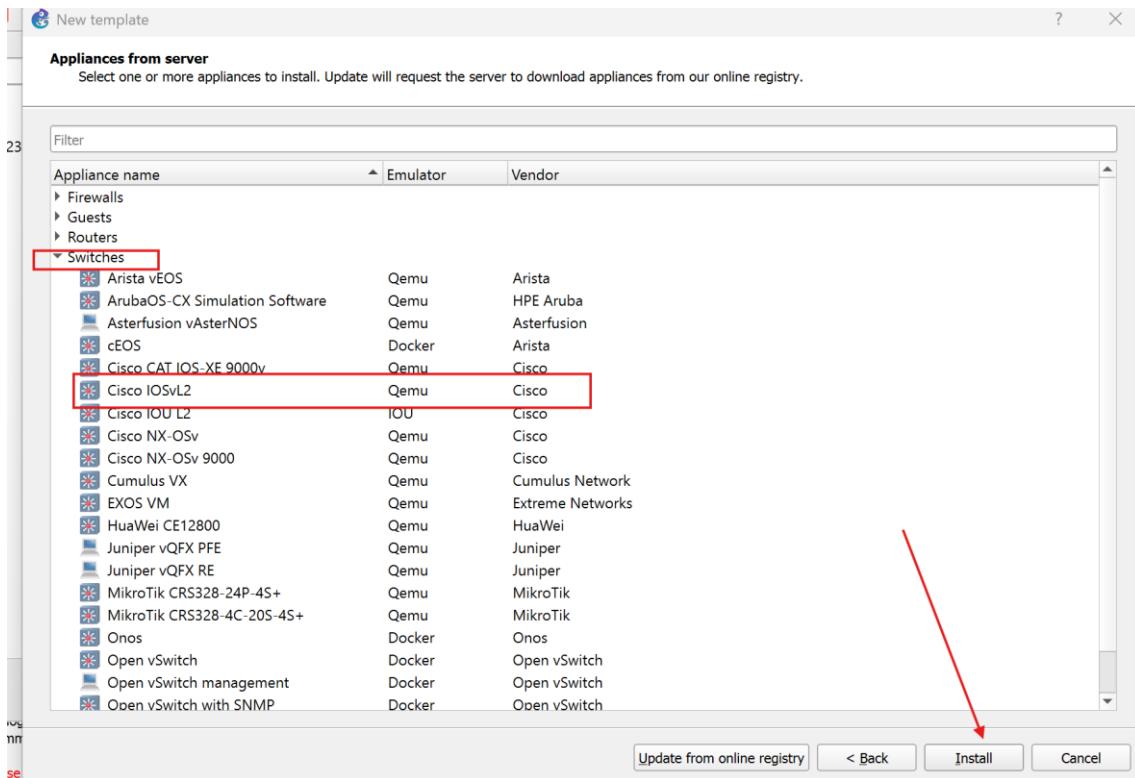
Se hace clic en **Next**.



## 2. Seleccionar el switch

En la categoría Switches, se busca y selecciona: **Cisco IOSvL2**.

Se hace clic en **Install**.

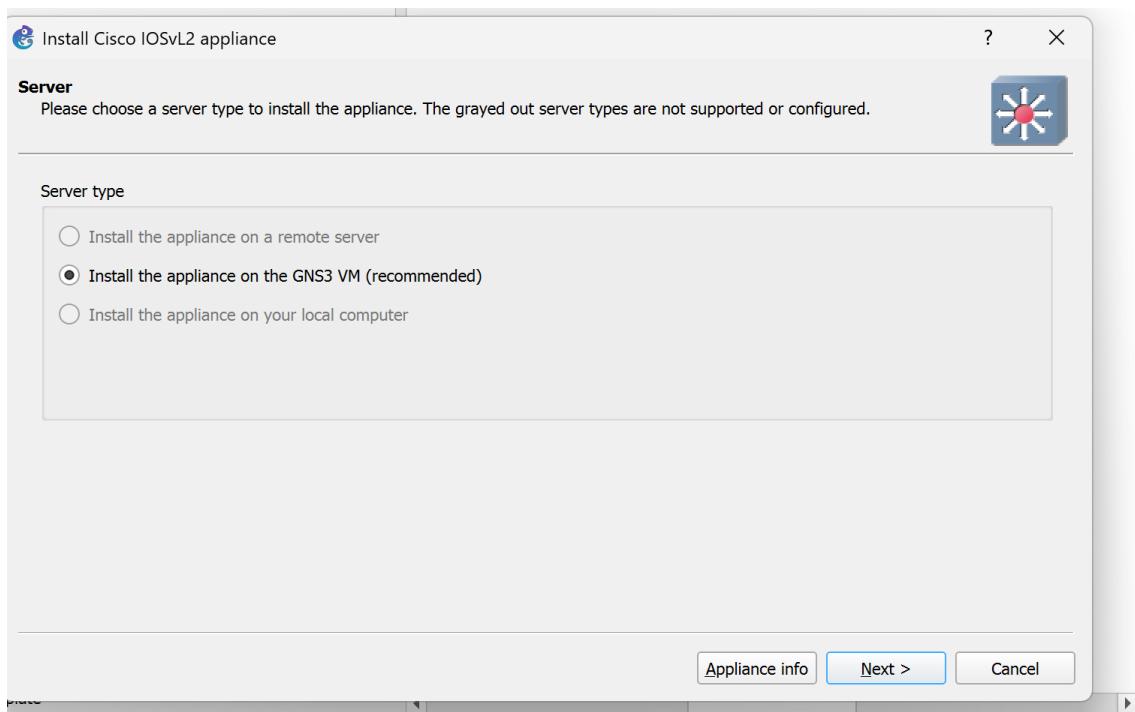


### 3. Seleccionar el servidor para instalar

Se Escoge:

Install the appliance on the GNS3 VM (recommended)

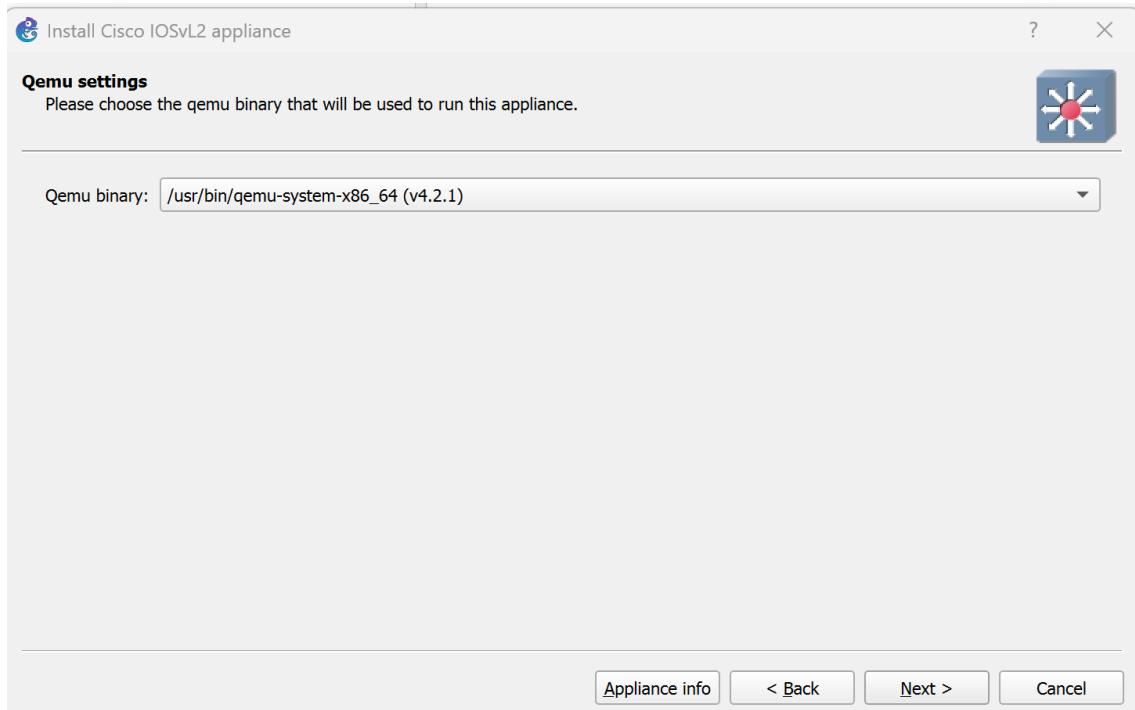
Se hace clic en **Next**.



#### 4. Confirmar binario de QEMU

Se verifica que el campo Qemu binary esté completo (por ejemplo: /usr/bin/qemu-system-x86\_64).

Se presiona **Next**.



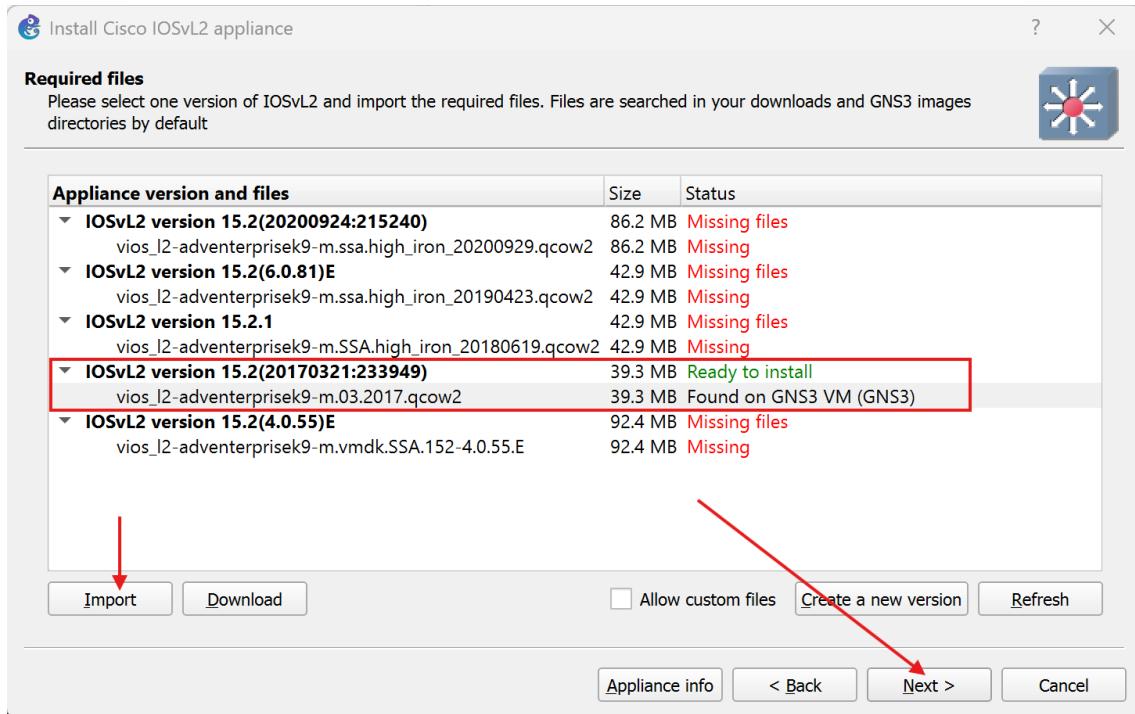
#### 5. Cargar o detectar la imagen

Se elige una versión disponible (por ejemplo: **IOSvL2 15.2 (20170321:233949)**).

Si ya está cargada, se verá **Ready to install** o Found on GNS3 VM.

Si no está, se puede importar o **descargar la imagen .qcow2**.

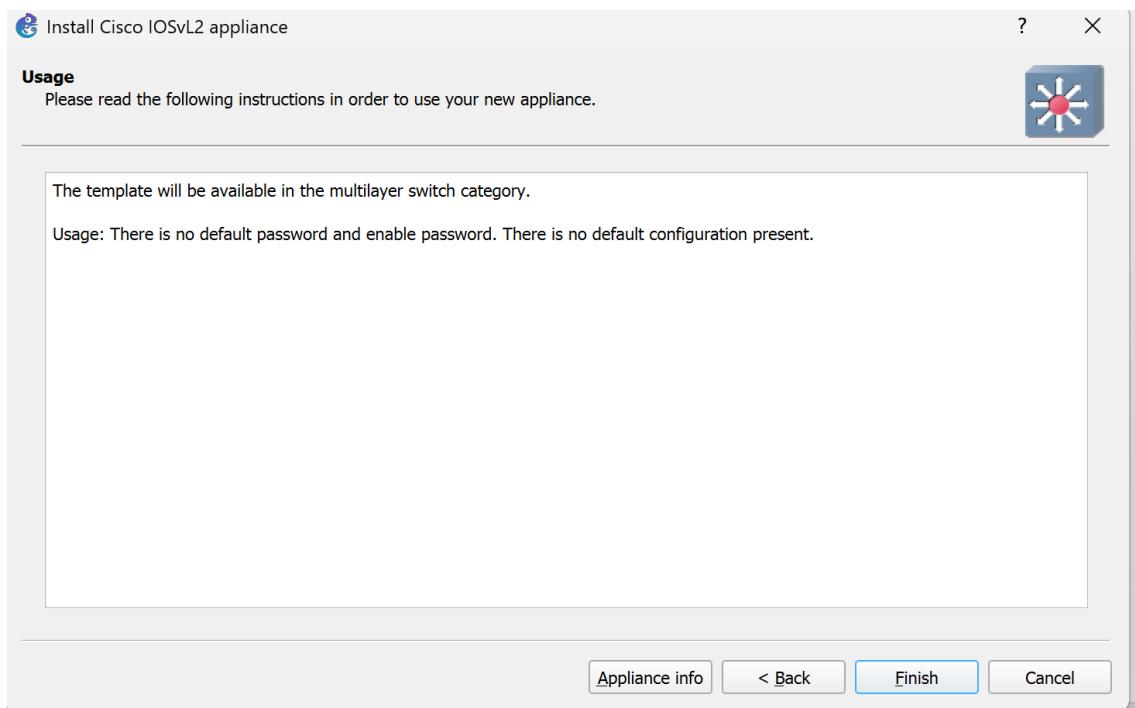
Haz clic en **Next**.



## 6. Finalizar la instalación

Aparecerá un resumen que indica que se agregará a la categoría "**multilayer switch**".

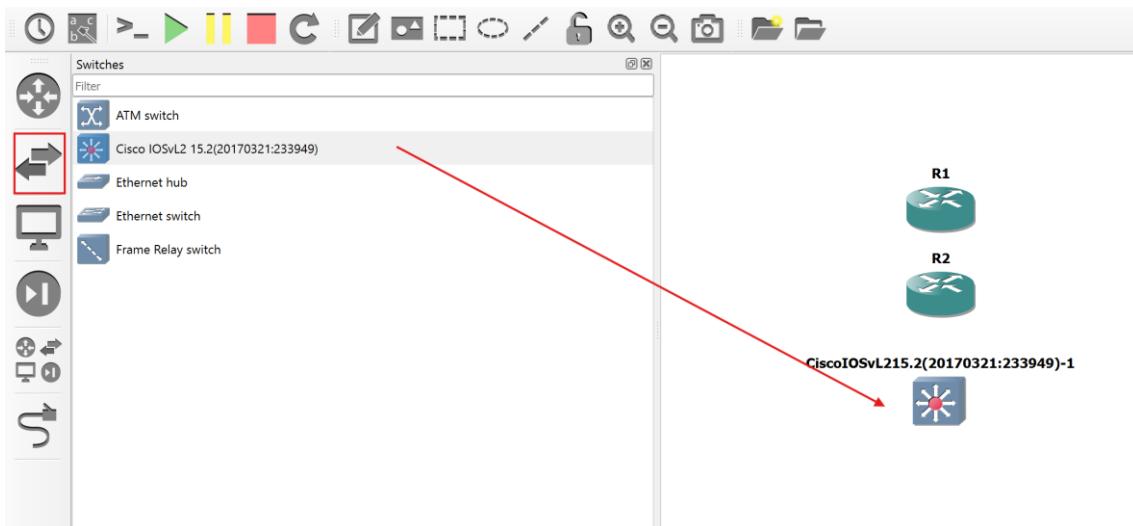
Se pulsa en **Finish**.



### 3.9.4 Añadir el switch al proyecto

En la lista de dispositivos ahora aparece el router c7200.

Se arrastra dos veces al área de trabajo para crear SW.



### 3.10 Pasos para añadir una Cloud en GNS3

En GNS3, busca en el panel izquierdo la categoría End devices.

Se selecciona **Cloud** (ícono de nube).

Se arrastra al área de trabajo.

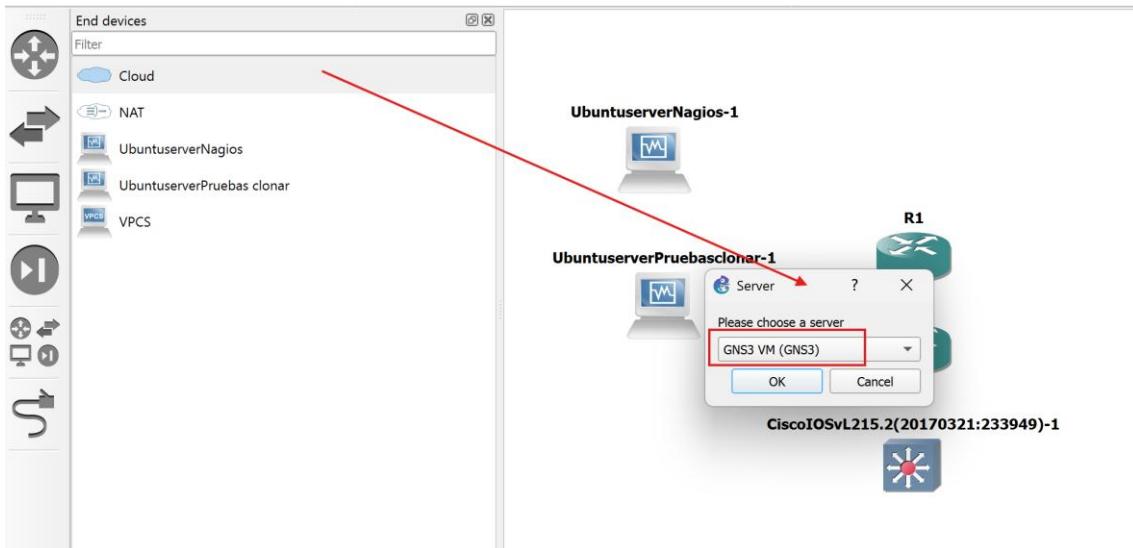
Se selecciona el servidor:

Al soltar la nube, aparecerá una ventana emergente que dice:  
**"Please choose a server"**

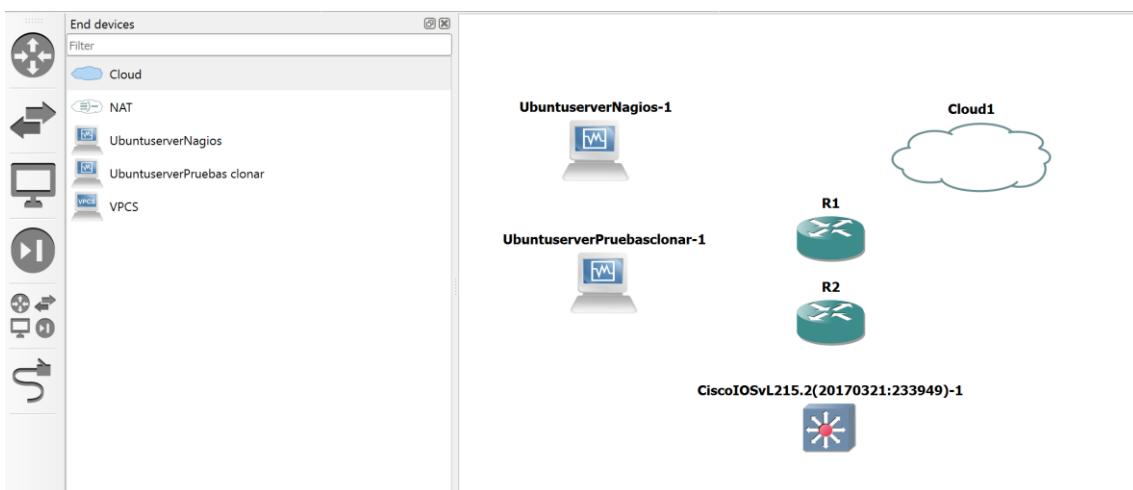
Se selecciona:

**GNS3 VM (GNS3)**

Se hace clic en **OK**.



Se puede observar en la topología tras finalizar.



### 3.11 Conexión física de la topología

Se conecta los dispositivos usando la herramienta de conexión:

| Dispositivo origen | Interfaz   | Conectado a   | Interfaz destino | Descripción                |
|--------------------|------------|---------------|------------------|----------------------------|
| Router R1          | F0/0       | Switch1       | G0/0             | Trunk VLAN 1 y VLAN 2      |
| Router R1          | F1/0       | Server Syslog | eth0             | Red de monitoreo (Syslog)  |
| Router R1          | F1/1       | Server Nagios | eth0             | Red de monitoreo (Nagios)  |
| Router R1          | S3/0 (DCE) | Router R2     | S1/0             | Enlace serial privado      |
| Switch1            | G0/1       | PC1           | eth0             | VLAN 1                     |
| Switch1            | G0/2       | PC2           | eth0             | VLAN 2                     |
| Router R2          | F0/0       | Cloud1        | -                | Conexión a Internet (DHCP) |

### 3.13 Conexión de la Topología de GNS3 a Internet mediante Interfaz Loopback y Adaptador Bridge

Para permitir que los dispositivos dentro de la topología de GNS3 tengan acceso a Internet, se configura una interfaz de red virtual tipo *loopback* en el sistema operativo anfitrión (Windows). Esta interfaz actúa como puente entre la red virtual simulada y la red física real.

#### 3.13.1 Creación y configuración de la interfaz Loopback

Instalación del adaptador Loopback:

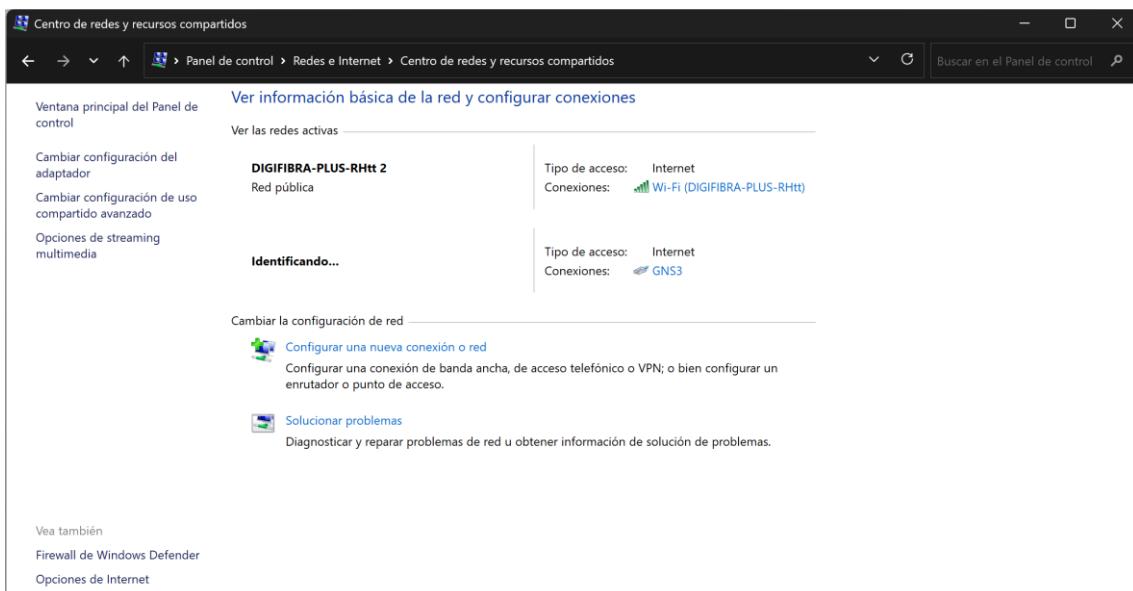
En Windows, se utiliza el asistente de instalación de hardware heredado (hdwwiz.exe) para agregar manualmente el Adaptador de bucle invertido de Microsoft.

### Activación automática de configuración IP:

Una vez instalada, no es necesario asignar manualmente una IP, ya que al compartir la conexión a Internet desde la interfaz Wi-Fi a la loopback, Windows asigna automáticamente una dirección IP del rango 192.168.137.0/24. Por ejemplo:

- Dirección IP: 192.168.137.1 (asignada a la interfaz de loopback)
- Máscara de subred: 255.255.255.0
- Puerta de enlace: no requerida manualmente
- DNS: asignado automáticamente (e.g., 8.8.8.8)

Esta interfaz actúa como puente entre la red física e Internet para los dispositivos en GNS3.



**Imagen 6 Vista del Centro de redes y recursos compartidos mostrando la red Wi-Fi principal y la red virtual “GNS3” con acceso a Internet.**

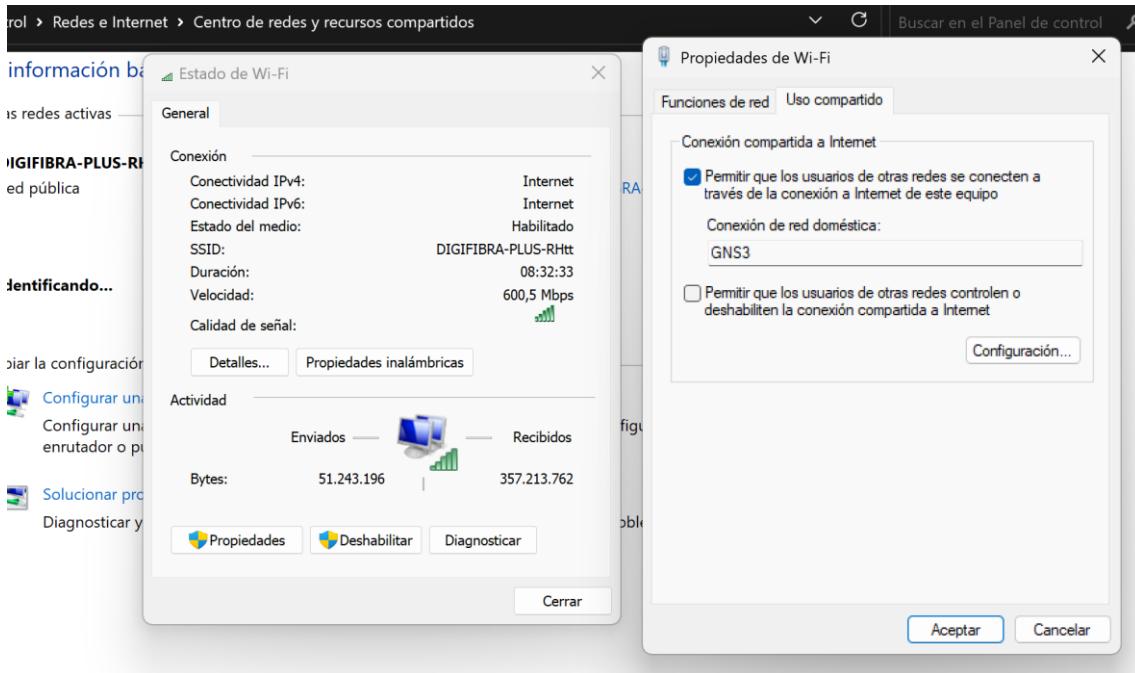
### 3.13.2 Compartir Internet con GNS3

Se accede a las propiedades de la conexión Wi-Fi desde el *Centro de redes y recursos compartidos*.

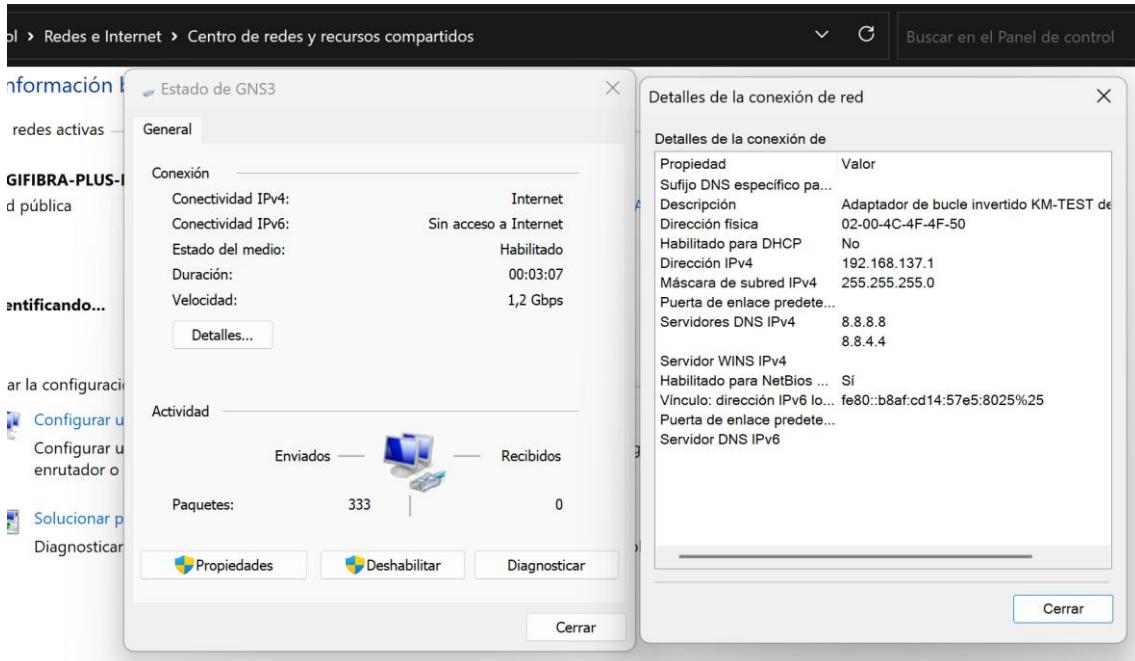
En la pestaña Uso compartido, se habilita la opción:

"Permitir que los usuarios de otras redes se conecten a través de la conexión a Internet de este equipo".

Como Conexión de red doméstica, se selecciona la interfaz "GNS3" (correspondiente al adaptador Loopback instalado).



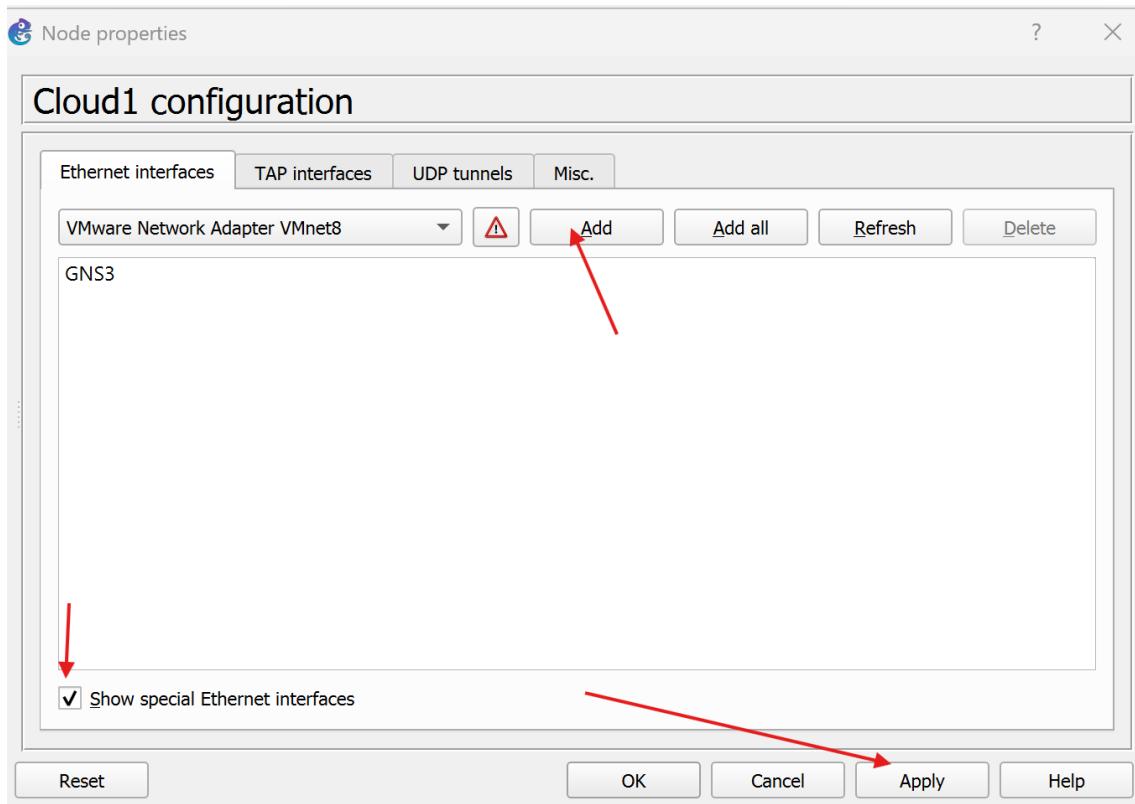
**Imagen 7 Propiedades del adaptador Wi-Fi mostrando el uso compartido activado y vinculado a la red virtual GNS3.**



### 3.13.3 Configuración en GNS3: Añadir Cloud y vincular interfaz Loopback

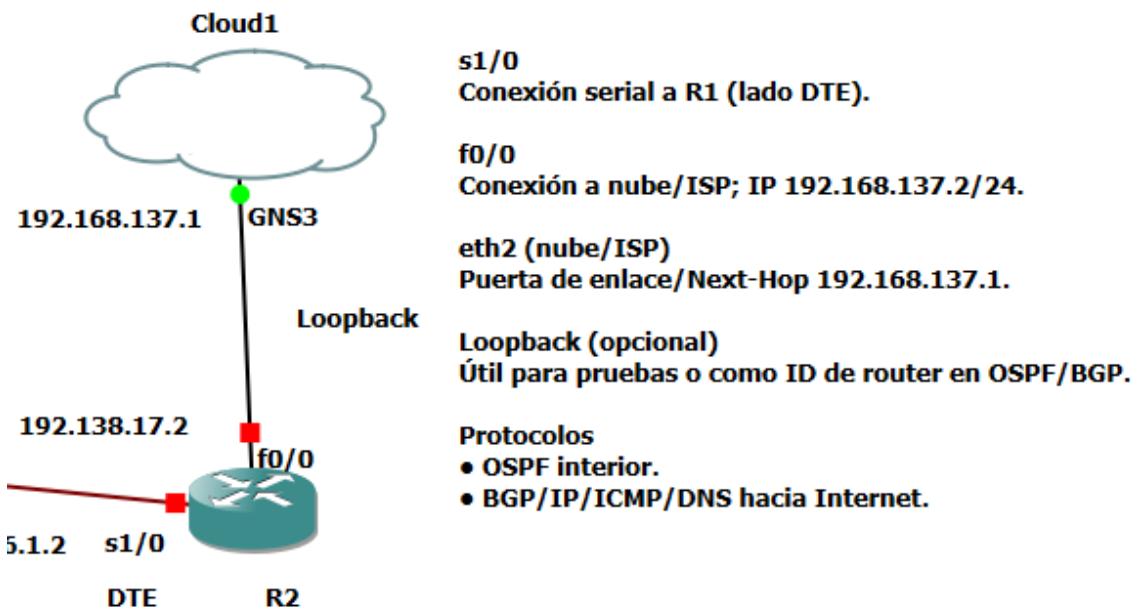
En GNS3 se añade un nodo de tipo Cloud desde la categoría *End Devices*.

En las propiedades de la Cloud, se selecciona la interfaz correspondiente a la loopback creada (identificada por su nombre o dirección IP).



Se conecta mediante cable Ethernet la Cloud al router o switch deseado.

**Conexión Internet  
(Protocolos: BGP, IP, ICMP, DNS)**



*Imagen 8 Conexión a Internet*

### 3.13.4 Configuración del router en GNS3

En el router conectado a la Cloud se configura la interfaz que apunta a la red de la loopback. Ejemplo:

```
interface FastEthernet0/0
ip address 192.168.137.2 255.255.255.0
no shutdown
```

Y se establece una ruta por defecto hacia la IP de la interfaz Loopback:

```
ip route 0.0.0.0 0.0.0.0 192.168.137.1
```

### 3.13.5 Verificación de conectividad

Una vez completada la configuración, se realizan pruebas de conectividad desde el router, PC virtual o cualquier otro nodo de GNS3, utilizando, por ejemplo:

```
ping 8.8.8.8
```

Una respuesta satisfactoria indica que los dispositivos dentro de GNS3 están accediendo correctamente a Internet mediante el adaptador Loopback.

```
R2#
R2#PING 8.8.8.8
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 8.8.8.8, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/29/36 ms
R2#
```

*Imagen 9 Estado y configuraConexión a Internetción de la interfaz “GNS3”, con IP estática 192.168.137.1 y DNS públicos de Google (8.8.8.8).*

## 3.14 Configuración Server Syslog

En este proyecto se implementa un sistema centralizado de gestión de logs mediante un servidor **Syslog** en un entorno Linux. La finalidad de esta configuración es recopilar, almacenar y supervisar los mensajes de registro (logs) generados tanto por equipos clientes (PCs) como por dispositivos de red (routers). Esta práctica mejora la seguridad, facilita la auditoría y permite una mejor respuesta ante incidentes, al centralizar los eventos del sistema en un único punto de monitoreo.

Se utiliza el servicio rsyslog, ampliamente soportado en sistemas Unix/Linux, para recibir y registrar eventos remotos por el puerto UDP 514. Asimismo, se preparan los equipos cliente para enviar sus eventos de forma automática, incluso tras reinicios. La solución también contempla la recolección de mensajes provenientes de routers Cisco configurados para enviar sus logs al mismo servidor Syslog.

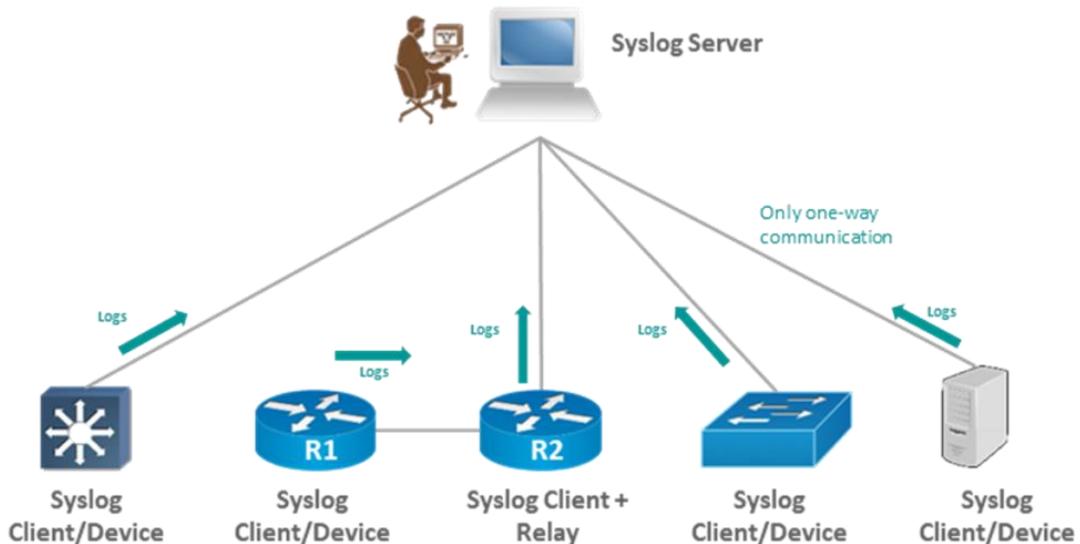


Imagen 10 Funcionamiento de Syslog

### 3.14.1 Instalación y Activación del Servicio Syslog

Se aplican los siguientes comandos de configuración.

#### Actualizar lista de paquetes:

```
sudo apt update
```

#### Instalar Rsyslog:

```
sudo apt install -y rsyslog
```

#### Desenmascarar, habilitar y arrancar el servicio:

```
sudo systemctl unmask rsyslog.service
```

```
sudo systemctl enable rsyslog.service
```

```
sudo systemctl start rsyslog.service
```

#### Verificar que el servicio está activo:

```
sudo systemctl status rsyslog.service
```

Debe mostrar: "active (running)"

```
alumno@UbuntuServerPruebas: ~$ sudo systemctl restart syslog-ng
alumno@UbuntuServerPruebas: ~$ sudo systemctl status syslog-ng
● syslog-nginx.service - System Logger Daemon
    Loaded: loaded (/lib/systemd/system/syslog-nginx.service; enabled; vendor preset: enabled)
      Active: active (running) since Sat 2025-04-26 20:39:53 UTC; 11s ago
        Docs: man:syslog-nginx(8)
       Main PID: 1517 (syslog-nginx)
          Tasks: 2 (limit: 24064)
         Memory: 3.5M
            CPU: 26ms
           CGrou...  
1517 /usr/sbin/syslog-nginx -F
```

Imagen 11 Servicio activo de Syslog Server

### 3.14.2 Configurar recepción de logs remotos

Se aplican los siguientes comandos de configuración.

#### Crear directorio para almacenar logs remotos:

```
sudo mkdir -p /var/log/remote
```

```
sudo chown syslog:adm /var/log/remote
```

```
sudo chmod 750 /var/log/remote
```

### 3.14.3 Editar configuración de rsyslog para escuchar por red

Se aplican los siguientes comandos de configuración.

#### Crear el archivo:

```
sudo nano /etc/rsyslog.d/01-remote.conf
```

Agregar las siguientes líneas:

#### Escuchar por UDP en el puerto 514:

```
module(load="imudp")
input(type="imudp" port="514")
```

#### Registrar logs remotos en subdirectorios por IP/host:

```
template(name="RemoteLogs" type="string"
        string="/var/log/remote/%HOSTNAME%/%PROGRAMNAME%.log")
.* ?RemoteLogs
& stop
```

### **3.14.4 Configurar Firewall**

Se aplican los siguientes comandos de configuración.

#### **Abrir el puerto 514/UDP:**

```
sudo ufw allow 514/udp
```

```
sudo ufw reload
```

### **3.14.5 Reiniciar rsyslog para aplicar cambios**

Se aplican los siguientes comandos de configuración.

```
sudo systemctl daemon-reexec
```

```
sudo systemctl restart rsyslog.service
```

### **3.14.6 Configuración de Cliente (PCs Linux)**

Se aplican los siguientes comandos de configuración.

#### **Instalar Rsyslog:**

```
sudo apt update
```

```
sudo apt install -y rsyslog
```

#### **Configurar envío de logs al servidor Syslog:**

```
echo '.*.* @10.10.11.2:514' | sudo tee /etc/rsyslog.d/10-remote.conf
```

#### **Reiniciar el servicio:**

```
sudo systemctl restart rsyslog.service
```

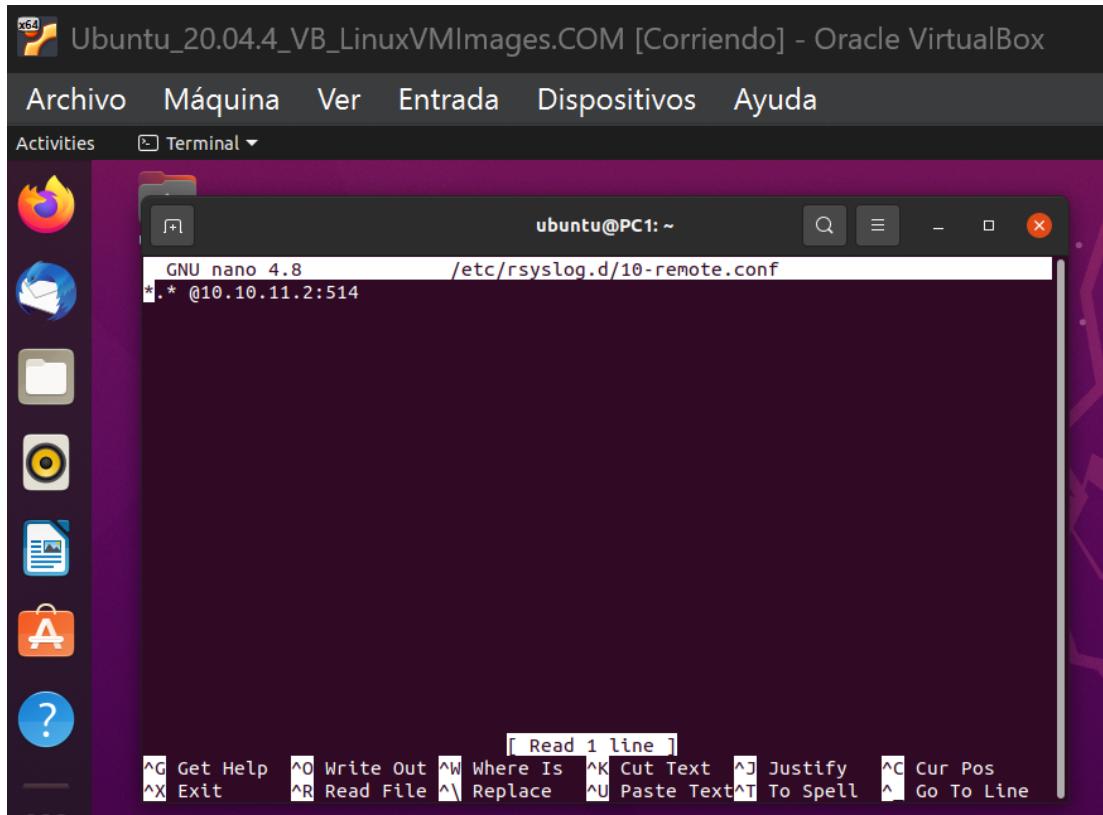


Imagen 12 Configuración de Cliente PC1 (Syslog)

### 3.14.7 Verificar persistencia tras reinicio

Se aplican los siguientes comandos de configuración.

Reiniciar el sistema:

```
sudo reboot
```

Tras reiniciar, comprobar IP y estado de Rsyslog:

```
ip a
```

```
ip route
```

```
systemctl status rsyslog
```

Enviar mensaje de prueba:

```
logger -p user.info "Post-boot desde $(hostname)"
```

### 3.14.8 Configuración en Routers Cisco

Se aplican los siguientes comandos de configuración.

```
R1(config)# logging host 10.10.11.2 transport udp port 514
```

```
R1(config)# logging trap informational
```

R1(config)# no logging console

### 3.14.9 Verificación en el Servidor Syslog

Se aplican los siguientes comandos de configuración.

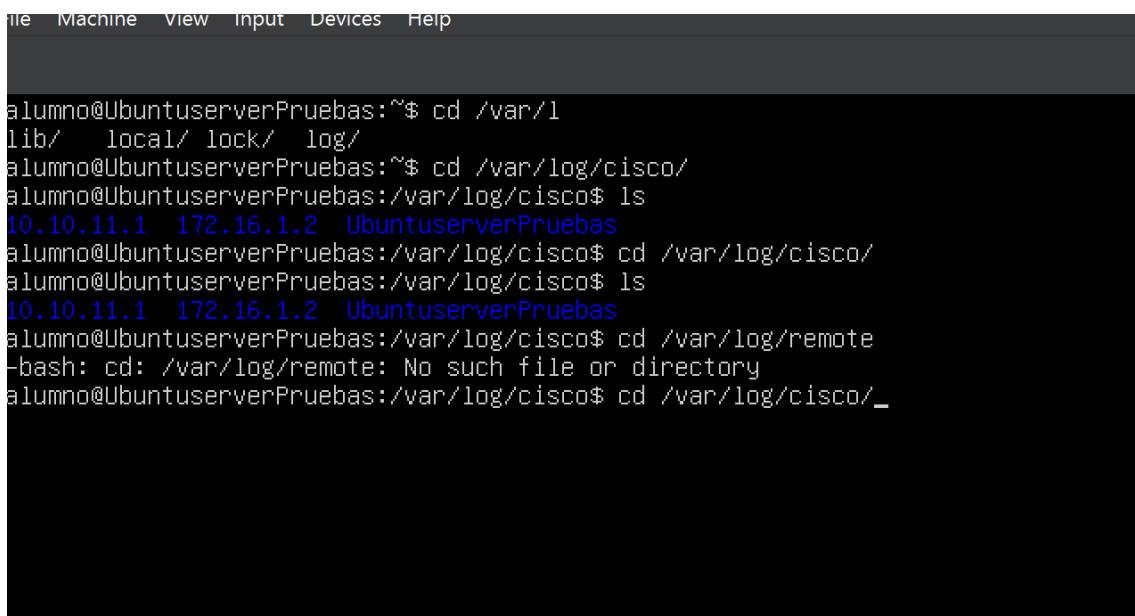
#### Verificar si se crean directorios por cliente:

ls /var/log/remote/

ls /var/log/remote/PC1

#### Ver logs en vivo:

sudo tail -f /var/log/remote/PC1/user.log



```
alumno@UbuntuserverPruebas:~$ cd /var/1
lib/ local/ lock/ log/
alumno@UbuntuserverPruebas:~$ cd /var/log/cisco/
alumno@UbuntuserverPruebas:/var/log/cisco$ ls
10.10.11.1 172.16.1.2 UbuntuserverPruebas
alumno@UbuntuserverPruebas:/var/log/cisco$ cd /var/log/cisco/
alumno@UbuntuserverPruebas:/var/log/cisco$ ls
10.10.11.1 172.16.1.2 UbuntuserverPruebas
alumno@UbuntuserverPruebas:/var/log/cisco$ cd /var/log/remote
-bash: cd: /var/log/remote: No such file or directory
alumno@UbuntuserverPruebas:/var/log/cisco$ cd /var/log/cisco/_
```

Imagen 13 Logs del Syslog

### 3.14.10 Generar Logs para Pruebas

Se aplican los siguientes comandos en los PCs Linux.

#### # AUTHPRIV: Intentos de sudo fallidos

```
for i in {1..50}; do
    sudo -k
    sudo ls /root &>/dev/null
done
```

**# DAEMON: Reinicios del servicio cron**

```
for i in {1..50}; do  
    sudo systemctl restart cron.service  
done
```

**# SYSLOG: Mensajes genéricos**

```
for i in {1..200}; do  
    logger "SYSLOG test #\$i desde $(hostname)"  
done
```

**# USER: Mensajes con facility user**

```
for i in {1..200}; do  
    logger -p user.info "USER test #\$i desde $(hostname)"  
done
```

**Para verificar los logs generados:**

```
# Ver archivos generados por cliente  
ls /var/log/remote/PC1/  
# → authpriv.log daemon.log syslog.log user.log
```

**3.15 Copiar los datos a mi pc mediante un túnel SSH reverse**

**¿Por qué GNS3 no permite el uso directo de adaptadores de red?**

GNS3 crea una **topología de red virtual** aislada del sistema anfitrión por defecto, lo que significa que los nodos (como los PCs y servidores Ubuntu dentro del laboratorio) **no tienen acceso directo a la red física** del host, a menos que se configuren correctamente adaptadores tipo *Cloud* o interfaces TAP/NIO.

Sin embargo, por limitaciones del sistema o permisos del usuario (especialmente en Windows), no siempre se pueden usar estos adaptadores de forma estable. Por esta razón, es común recurrir a **túneles SSH reversos o NAT en routers Cisco** dentro de GNS3 para exponer servicios como SSH hacia el exterior (host real).

## ¿Qué es un túnel SSH reverso?

Un **túnel SSH reverso** permite redirigir un puerto desde una máquina remota (como una VM en GNS3) hacia el sistema anfitrión o un tercero, facilitando el acceso incluso si esa máquina está detrás de NAT o no tiene IP pública directa.

En este caso, lo que se hace es un **NAT estático en el router R2** para que el puerto 2222 de su interfaz externa se redirija al puerto 22 (SSH) del servidor Ubuntu que corre dentro de GNS3.

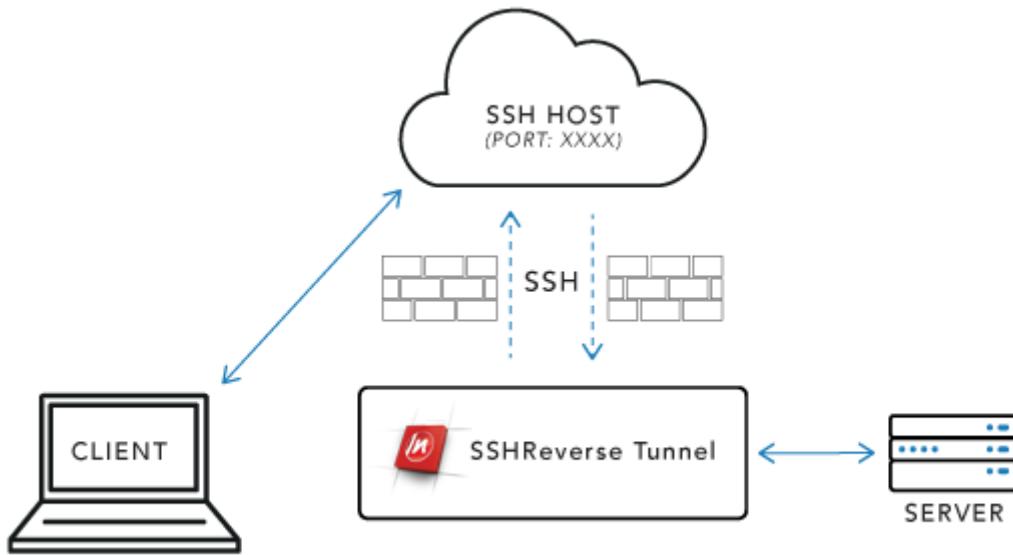


Imagen 14 Túnel SSH reverso

### 3.15.1 Pasos para exportar logs desde Ubuntu-Server a Windows

En R2, configura **NAT para exponer el puerto 22 del Ubuntu-Server (10.10.11.2) al exterior a través del puerto 2222**:

```
R2(config)# ip nat inside source static tcp 10.10.11.2 22 interface FastEthernet0/0 2222
R2(config)# interface Serial1/0
R2(config-if)# ip nat inside
R2(config)# interface FastEthernet0/0
R2(config-if)# ip nat outside
```

```
R2(config)#
R2(config)#ip nat inside source static tcp 10.10.11.2 22 interface FastEtherne
R2(config)#[
```

Imagen 15 NAT para exponer el puerto 22 del Ubuntu-Server (10.10.11.2) al exterior a través del puerto 2222

### Habilitar autenticación por contraseña en Ubuntu-Server:

Se asegura que SSH permita autenticación por contraseña (si no se ha configurado aún):

```
sudo sed -i 's/^#PasswordAuthentication no/PasswordAuthentication yes/' /etc/ssh/sshd_config
```

```
sudo sed -i 's/^#PasswordAuthentication no/PasswordAuthentication yes/' /etc/ssh/sshd_config
```

```
sudo systemctl restart ssh
```

### Comprimir los logs del servidor:

Desde Ubuntu-Server, se genera un archivo .tar.gz con todos los logs recopilados:

```
sudo tar czvf /home/alumno/cisco_logs.tar.gz -C /var/log remote
```

```
sudo chown alumno:alumno /home/alumno/cisco_logs.tar.gz
```

```
ls -lh /home/alumno/cisco_logs.tar.gz
```

### Conectarse desde tu PC al servidor usando SSH:

```
ssh -p 2222 alumno@192.168.137.2
```

# Usuario: alumno, y la contraseña: [la establecida en Ubuntu].

```
C:\Windows\System32>ssh -p 2222 alumno@192.168.137.2
alumno@192.168.137.2's password:
Welcome to Ubuntu 22.04.5 LTS (GNU/Linux 5.15.0-126-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Mon May 12 09:31:26 AM UTC 2025

 System load:  0.05           Processes:          154
 Usage of /:   17.0% of 24.44GB  Users logged in:     1
 Memory usage: 1%            IPv4 address for enp0s3: 10.10.11.2
 Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

136 updates can be applied immediately.
88 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

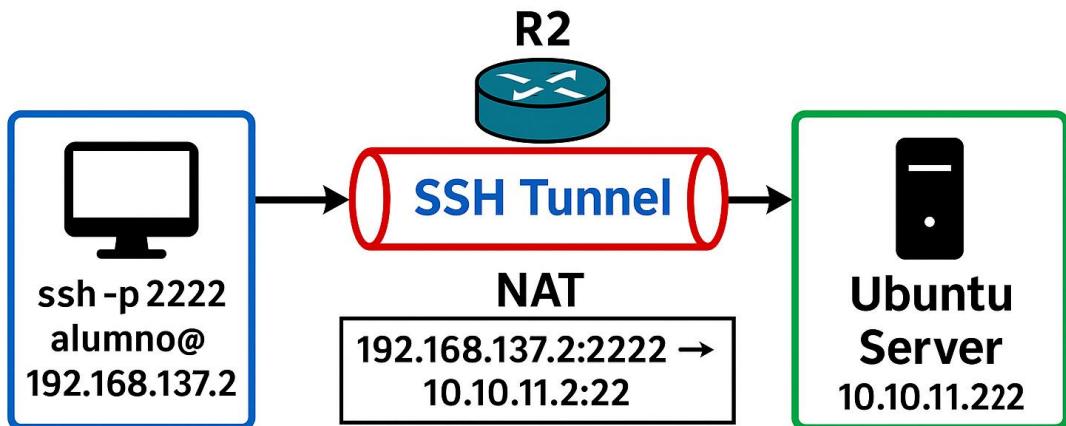
2 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

Last login: Mon May 12 09:22:53 2025
alumno@UbuntuServerPruebas:~$
```

*Imagen 16 Conexión del túnel establecida con el servidor*

Mediante la siguiente imagen se observa lo ejecutado:



#### Descargar el archivo a Windows:

##### Opción A: Usar SCP desde PowerShell (Windows 10+)

Primero, se asegura tener el cliente OpenSSH activado:

- Se va a **Settings > Apps > Optional Feature**
- Se busca e instala **OpenSSH Client**

Luego en PowerShell:

```
scp -P 2222 alumno@192.168.137.2:/home/alumno/cisco_logs.tar.gz  
C:\Users\Carla\Desktop\
```

##### Opción B: Usar WinSCP (Interfaz gráfica)

1. Se descarga e instala [WinSCP](#).
2. Se configura una nueva sesión:
  - **Protocol:** SFTP
  - **Host:** 192.168.137.2 (o 10.10.11.2 si vas directo)
  - **Port:** 2222 (o 22 si directo)
  - **Username:** alumno
3. Se conecta, navega a /home/alumno/, y se **arrastra el archivo cisco\_logs.tar.gz** al Escritorio.

### 3.16 Configuración Server Nagios

En este proyecto se implementa un servidor Nagios como solución centralizada de monitorización para la infraestructura virtualizada. La finalidad de esta configuración es supervisar de forma continua el estado y rendimiento de todos los equipos y servicios críticos, detectando de manera temprana posibles incidencias y facilitando alertas automáticas al equipo de operaciones. Esta práctica mejora notablemente la visibilidad del entorno, optimiza los tiempos de respuesta ante anomalías y contribuye a garantizar la disponibilidad y calidad de los servicios desplegados.

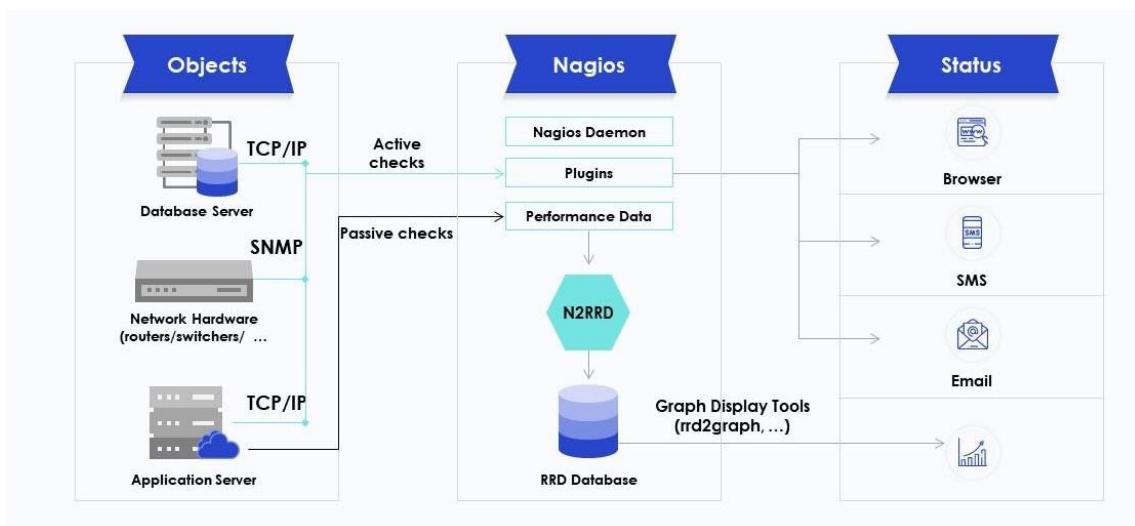


Imagen 17 Arquitectura de Nagios

#### Servicios monitorizados por host

Cada equipo monitorizado incluirá supervisión de los siguientes componentes:

- **Current Load:** carga del sistema
- **Current Users:** usuarios conectados
- **Disk Space:** uso del disco
- **HTTP:** acceso a servicios web
- **SSH:** disponibilidad de acceso remoto
- **Total Processes:** cantidad de procesos activos
- *(Opcional: SNMP u otros servicios avanzados)*

### 3.16.1 Preparación del sistema

Se aplican los siguientes comandos de configuración.

**Actualiza el sistema:**

```
sudo apt update && sudo apt upgrade -y
```

**Instala dependencias necesarias:**

```
sudo apt install -y apache2 php libapache2-mod-php build-essential libgd-dev  
unzip wget mailutils
```

### 3.16.2 Instalar Nagios Core

Se aplican los siguientes comandos de configuración.

**Crear usuario y grupo:**

```
alumno@nagios:~$ sudo useradd nagios  
sudo groupadd nagcmd  
sudo usermod -a -G nagcmd nagios  
sudo usermod -a -G nagcmd www-data  
useradd: user 'nagios' already exists  
groupadd: group 'nagcmd' already exists  
alumno@nagios:~$ |
```

**Descargar, compilar e instalar Nagios:**

```
cd /tmp
```

```
wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-  
4.4.14.tar.gz
```

```
tar -xzf nagios-4.4.14.tar.gz
```

```
cd nagios-4.4.14
```

```
./configure --with-command-group=nagcmd
```

```
make all
```

```
sudo make install
```

```
sudo make install-commandmode
```

```
sudo make install-init
```

```
sudo make install-config
```

```
sudo make install-webconf
```

### **3.16.3 Instalar plugins**

Se aplican los siguientes comandos de configuración.

```
cd /tmp
```

```
wget https://nagios-plugins.org/download/nagios-plugins-2.3.3.tar.gz
```

```
tar -xzf nagios-plugins-2.3.3.tar.gz
```

```
cd nagios-plugins-2.3.3
```

```
./configure --with-nagios-user=nagios --with-nagios-group=nagcmd
```

```
make
```

```
sudo make install
```

### **3.16.4 Acceso Web**

Se aplican los siguientes comandos de configuración.

**Crear usuario web:**

```
sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
```

**Habilitar CGI y reiniciar Apache:**

```
sudo a2enmod cgi
```

```
sudo systemctl restart apache2
```

### **3.16.5 Configuración personalizada**

Se aplican los siguientes comandos de configuración.

**Activar Nagios al inicio:**

```
sudo systemctl enable nagios
```

```
alumno@nagios:~$ sudo systemctl status nagios
● nagios.service - Nagios Core 4.4.14
   Loaded: loaded (/lib/systemd/system/nagios.service; enabled; vendor preset: enabled)
     Active: active (running) since Mon 2025-05-12 09:39:48 UTC; 7min ago
       Docs: http://www.nagios.org/documentation
 Main PID: 829 (nagios)
   Tasks: 14 (limit: 24063)
    Memory: 11.6M
      CPU: 396ms
     CGroup: /system.slice/nagios.service
             ├─829 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             ├─836 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─837 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─838 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─839 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─841 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─842 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─843 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─846 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─847 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─850 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─855 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─857 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             └─874 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

May 12 09:39:48 nagios nagios[829]: wproc: Registry request: name=Core Worker 855;pid=855
May 12 09:39:48 nagios nagios[829]: WARNING: The normal_check_interval attribute is deprecated and will be removed in future versions. Please use check_interval instead.
May 12 09:39:48 nagios nagios[829]: WARNING: The retry_check_interval attribute is deprecated and will be removed in future versions. Please use retry_interval instead.
May 12 09:39:48 nagios nagios[829]: Successfully launched command file worker with pid 874
May 12 09:40:08 nagios nagios[829]: SERVICE ALERT: Router-R1:PING;CRITICAL;SOFT;1;CRITICAL - Network Unreachable (172.16.1.1)
May 12 09:41:08 nagios nagios[829]: SERVICE ALERT: Router-R1:PING;CRITICAL;SOFT;2;CRITICAL - Network Unreachable (172.16.1.1)
May 12 09:43:12 nagios nagios[829]: SERVICE ALERT: Router-R1:PING;CRITICAL;SOFT;3;CRITICAL - Network Unreachable (172.16.1.1)
May 12 09:45:12 nagios nagios[829]: SERVICE ALERT: Router-R1:PING;OK;SOFT;4;PING OK - Packet loss = 0%, RTA = 11.82 ms
May 12 09:46:58 nagios nagios[829]: SERVICE ALERT: PC1:PING;CRITICAL;SOFT;1;PING CRITICAL - Packet loss = 90%, RTA = 63.49 ms
May 12 09:46:58 nagios nagios[829]: SERVICE ALERT: PC1:PING;CRITICAL;SOFT;2;PING CRITICAL - Packet loss = 90%, RTA = 99.16 ms
alumno@nagios:~$
```

Imagen 18 Servicio activo de Nagios Server

A continuación, se detalla una breve descripción de la función de cada uno de esos ficheros de configuración de Nagios:

## 1. nagios.cfg

Archivo principal de configuración de Nagios. **Define la ruta y los parámetros globales del demonio** (niveles de log, intervalos de comprobación, directorios de objetos incluidos, etc.). Aquí es donde “**se habilitan**” (mediante las líneas cfg\_file=) todos los bloques de definición que luego Nagios cargará.

Se aplican los siguientes comandos de configuración.

### nagios.cfg – habilitar solo los archivos usados:

```
sudo nano /usr/local/nagios/etc/nagios.cfg
```

Se agrega o edita:

```
cfg_file=/usr/local/nagios/etc/objects/contacts.cfg
```

```
cfg_file=/usr/local/nagios/etc/objects/timeperiods.cfg
```

```
cfg_file=/usr/local/nagios/etc/objects/commands.cfg
```

```
cfg_file=/usr/local/nagios/etc/objects/templates.cfg
```

```
cfg_file=/usr/local/nagios/etc/objects/hosts.cfg
```

```
cfg_file=/usr/local/nagios/etc/objects/services.cfg
```

## 2. contacts.cfg

**Define los contactos o grupos de contacto** a los que Nagios enviará alertas y **notificaciones**. En este fichero se declaran cosas como:

- contact\_name
- email
- Métodos y escalado de notificaciones (por e-mail, SMS, etc.)
- A qué hosts y servicios están suscritos.

Se aplican los siguientes comandos de configuración al archivo como se observa en la siguiente imagen:

```
alumno@nagios:/usr/local/nagios/etc/objects$ ls
commands.cfg      commands-snmp.cfg    contacts.cfg.bak   hosts.cfg.bak     printer.cfg    services.cfg.bak  templates.cfg    timeperiods.cfg    topologia.cfg
commands.cfg.bak  contacts.cfg        hosts.cfg        localhost.cfg.bak  services.cfg   switch.cfg     templates.cfg.bak  timeperiods.cfg.bak  windows.cfg
alumno@nagios:/usr/local/nagios/etc/objects$ sudo nano contacts.cfg
alumno@nagios:/usr/local/nagios/etc/objects$ cat contacts.cfg
define contact {
    contact_name          nagiosadmin
    alias                 Nagios Admin
    service_notification_period 24x7
    host_notification_period 24x7
    service_notification_options w,u,c,r
    host_notification_options d,u,r
    service_notification_commands notify-service-by-email
    host_notification_commands notify-host-by-email
    email                carlautreracalderon@gmail.com
    use                  generic-contact
}

define contactgroup {
    contactgroup_name      admins
    alias                 Administradores de Nagios
    members               nagiosadmin
}
alumno@nagios:/usr/local/nagios/etc/objects$ |
```

## 3. timeperiods.cfg

Contiene **definiciones de ventanas de tiempo** (time periods) durante las cuales Nagios **realizará comprobaciones o enviará notificaciones**. Por ejemplo:

- “Horario laboral” (L-V 09:00–18:00)
- “Fin de semana”
- “Festivos”

Estas definiciones se referencian luego en los hosts, servicios o contactos para controlar **cuándo se monitorea o alerta**.

Se aplican los siguientes comandos de configuración al archivo como se observa en la siguiente imagen:

```
alumno@nagios:/usr/local/nagios/etc/objects$ cat timeperiods.cfg
define timeperiod {
    timeperiod_name 24x7
    alias            24 Hours A Day, 7 Days A Week
    sunday          00:00-24:00
    monday          00:00-24:00
    tuesday         00:00-24:00
    wednesday       00:00-24:00
    thursday        00:00-24:00
    friday          00:00-24:00
    saturday         00:00-24:00
}
alumno@nagios:/usr/local/nagios/etc/objects$ |
```

#### 4. templates.cfg

Define **plantillas reutilizables de hosts y servicios** (host & service templates). Permiten **centralizar valores comunes** (intervalos, parámetros de notificación, umbrales, etc.) para no repetirlos en cada objeto.

Se aplican los siguientes comandos de configuración al archivo como se observa en la siguiente imagen:

```

alumno@nagios:/usr/local/nagios/etc/objects$ cat templates.cfg
define host {
    name                  generic-host
    check_period          24x7
    check_interval        5
    retry_interval        1
    max_check_attempts    5
    notification_period   24x7
    notification_interval 30
    notification_options  d,u,r
    contact_groups        admins
    register              0
}

define service {
    name                  generic-service
    active_checks_enabled 1
    passive_checks_enabled 1
    check_period          24x7
    max_check_attempts    4
    normal_check_interval 5
    retry_check_interval  1
    notification_interval 30
    notification_period   24x7
    notification_options  w,u,c,r
    contact_groups        admins
    register              0
}
define contact {
    name                  generic-contact
    service_notification_period 24x7
    host_notification_period 24x7
    service_notification_options w,u,c,r
    host_notification_options d,u,r
    service_notification_commands notify-service-by-email
    host_notification_commands  notify-host-by-email
    register              0
}
alumno@nagios:/usr/local/nagios/etc/objects$ |

```

## 5. commands.cfg

Agrupa las **definiciones de comandos de comprobación** (check\_command) y de notificación (notify\_command).

Se aplican los siguientes comandos de configuración al archivo como se observa en la siguiente imagen:

```
lunes@nagios:/usr/local/nagios/etc/objects$ cat commands.cfg
define command {
    command_name    check_ping
    command_line    $USER1$/check_ping -H $HOSTADDRESS$ -w $ARG1$ -c $ARG2$
}

define command {
    command_name    check_http
    command_line    $USER1$/check_http -H $HOSTADDRESS$ 
}

define command {
    command_name    check_ssh
    command_line    $USER1$/check_ssh $HOSTADDRESS$ 
}

define command {
    command_name    check_users
    command_line    $USER1$/check_users -w $ARG1$ -c $ARG2$ 
}

define command {
    command_name    check_load
    command_line    $USER1$/check_load -w $ARG1$ -c $ARG2$ 
}

define command {
    command_name    check_disk
    command_line    $USER1$/check_disk -w $ARG1$ -c $ARG2$ -p /
}

define command {
    command_name    check_procs
    command_line    $USER1$/check_procs -w $ARG1$ -c $ARG2$ 
}

define command {
    command_name    notify-service-by-email
    command_line    /usr/bin/printf "%b" "Notificación: $SERVICEDESC$ en $HOSTNAME$ está $SERVICESTATE$\n\n$OUTPUT$" | /usr/bin/mail -s """ $NOTIFICATIONTYPE$ - $HOSTNAME$/$SERVICEDESC$ está $SERVICESTATE$ """ $CONTACTEMAILS$ 
}

define command {
    command_name    notify-host-by-email
    command_line    /usr/bin/printf "%b" "Notificación de host: $HOSTNAME$ está $HOSTSTATE$\n\n$OUTPUT$" | /usr/bin/mail -s """ $NOTIFICATIONTYPE$ - Host $HOSTNAME$ está $HOSTSTATE$ """ $CONTACTEMAILS$ 
}

lunes@nagios:/usr/local/nagios/etc/objects$
```

## 6. hosts.cfg

Contiene las **definiciones de cada host que quieras supervisar**:

- **host\_name**
- **Dirección IP o FQDN (address)**
- **Plantilla base (use)**
- **Grupos de hosts**
- **Timeperiods o contactos específicos**

A partir de aquí Nagios sabe dónde debe dirigir las comprobaciones.

Se aplican los siguientes comandos de configuración al archivo como se observa en la siguiente imagen:

```
alumno@nagios:/usr/local/nagios/etc/objects$ cat hosts.cfg
# Servidor Nagios
define host {
    use                      generic-host
    host_name                Server-Nagios
    alias                     Servidor Nagios (Docker)
    address                  10.10.10.2
}

# Servidor Syslog
define host {
    use                      generic-host
    host_name                Server-Syslog
    alias                     Servidor Syslog (Docker)
    address                  10.10.11.2
}

# Router R1 (Cisco)
define host {
    use                      generic-host
    host_name                Router-R1
    alias                     Router Principal R1
    address                  10.10.10.1
}

# Router R2 (Cisco)
define host {
    use                      generic-host
    host_name                Router-R2
    alias                     Router Secundario R2
    address                  172.16.1.2
}

# PC1 (VLAN 1)
define host {
    use                      generic-host
    host_name                PC1
    alias                     Equipo Usuario VLAN 1
    address                  192.168.1.10
}

# PC2 (VLAN 2)
define host {
    use                      generic-host
    host_name                PC2
    alias                     Equipo Usuario VLAN 2
    address                  192.168.2.10
}
alumno@nagios:/usr/local/nagios/etc/objects$ |
```

## 7. services.cfg

Agrupa todas las **comprobaciones (“servicios”)** asociadas a cada host:

- A qué host aplica (**host\_name**)
- Qué comando ejecutar (**check\_command**)
- Intervalos de comprobación y reintentos
- Parámetros adicionales (por ejemplo, puertos HTTP/SSH, rutas de disco...)

Cada línea en este fichero es un servicio diferente que Nagios monitoriza.

Se aplican los siguientes comandos de configuración al archivo como se observa en la siguiente imagen:

```

alumno@nagios:/usr/local/nagios/etc/objects$ cat services.cfg
# -----
# 1. Servicios comunes: PING para todos los hosts
# -----
define service {
    use          generic-service
    host_name   Server-Nagios,Server-Syslog,Router-R1,Router-R2,PC1,PC2
    service_description  PING
    check_command    check_ping!100.0,20%!500.0,60%
}

# -----
# 2. Servicios avanzados en servidores (Nagios y Syslog)
# -----
define service {
    use          generic-service
    host_name   Server-Nagios,Server-Syslog
    service_description  SSH
    check_command    check_ssh
}

define service {
    use          generic-service
    host_name   Server-Nagios,Server-Syslog
    service_description  HTTP
    check_command    check_http
}

define service {
    use          generic-service
    host_name   Server-Nagios,Server-Syslog
    service_description  Current Load
    check_command    check_load!5.0,4.0,3.0!10.0,6.0,4.0
}

define service {
    use          generic-service
    host_name   Server-Nagios,Server-Syslog
    service_description  Current Users
    check_command    check_users!20!50
}

define service {
    use          generic-service
    host_name   Server-Nagios,Server-Syslog
    service_description  Disk Space
    check_command    check_disk!20%!10%
}

define service {
    use          generic-service
    host_name   Server-Nagios,Server-Syslog
    service_description  Total Processes
    check_command    check_procs!250!400
}

# -----
# 3. Servicios en routers (opcional SSH/HTTP)
# -----
define service {
    use          generic-service
    host_name   Router-R1,Router-R2
    service_description  SSH
    check_command    check_ssh
}

# Si tus routers tienen servidor web para gestión:
#define service {
#    use          generic-service
#    host_name   Router-R1,Router-R2
#    service_description  HTTP
#    check_command    check_http
#}

# -----
# 4. Servicios en PCs (ping + SSH si aplica)
# -----
define service {
    use          generic-service
    host_name   PC1,PC2
    service_description  SSH
    check_command    check_ssh
}

alumno@nagios:/usr/local/nagios/etc/objects$ |

```

## Verificar sintaxis:

Se comprueba la sintaxis mediante el siguiente comando:

```
sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
```

```
alumno@nagios:/usr/local/nagios/etc/objects$ sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
Nagios Core 4.4.14
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2023-08-01
License: GPL

Website: https://www.nagios.org
Reading configuration data...
  Read main config file okay...
  Read object config files okay...
  Running pre-flight check on configuration data...

Checking objects...
  Checked 22 services.
  Checked 6 hosts.
  Checked 0 host groups.
  Checked 0 service groups.
  Checked 1 contacts.
  Checked 1 contact groups.
  Checked 9 commands.
  Checked 1 time periods.
  Checked 0 host escalations.
  Checked 0 service escalations.
Checking for circular paths...
  Checked 6 hosts
  Checked 0 service dependencies
  Checked 0 host dependencies
  Checked 1 timperiods
Checking global event handlers...
Checking obsessive compulsive processor commands...
Checking misc settings...

Total Warnings: 0
Total Errors: 0

Things look okay - No serious problems were detected during the pre-flight check
alumno@nagios:/usr/local/nagios/etc/objects$ |
```

Imagen 19 Correcta configuración de los archivos Nagios definidos

## Reiniciar Nagios:

```
sudo systemctl restart nagios
```

### 3.16.6 Acceder a la interfaz web de Nagios

Una vez instalado y reiniciado Nagios correctamente, puedes acceder a la interfaz desde **cualquier navegador**.

- Desde el servidor Ubuntu

**http://localhost/nagios o http://127.0.0.1/nagios**

- Desde un equipo cliente

En tu navegador, accede a:

**http://<IP-del-servidor>/nagios**

Asegúrate de que ambos estén en la misma red (modo NAT con puertos mapeados o bridge).

### Credenciales:

Usuario: **nagiosadmin**

Contraseña: la definida con **htpasswd**

## 3.17 Acceso al panel de Nagios y la extracción de datos

En la configuración de R2, autoriza el rango de direcciones de la red Nagios (10.10.10.0/30):

! Red Nagios (10.10.10.0/30)

```
R2(config)# access-list 10 permit 10.10.10.0 0.0.0.3
```

### 3.17.1 Port-forwarding para acceder a Nagios desde el exterior

Se necesita acceder al interfaz web de Nagios (puerto 80) o por SSH (puerto 22) desde la IP pública de R2, configura NAT estático de la siguiente forma:

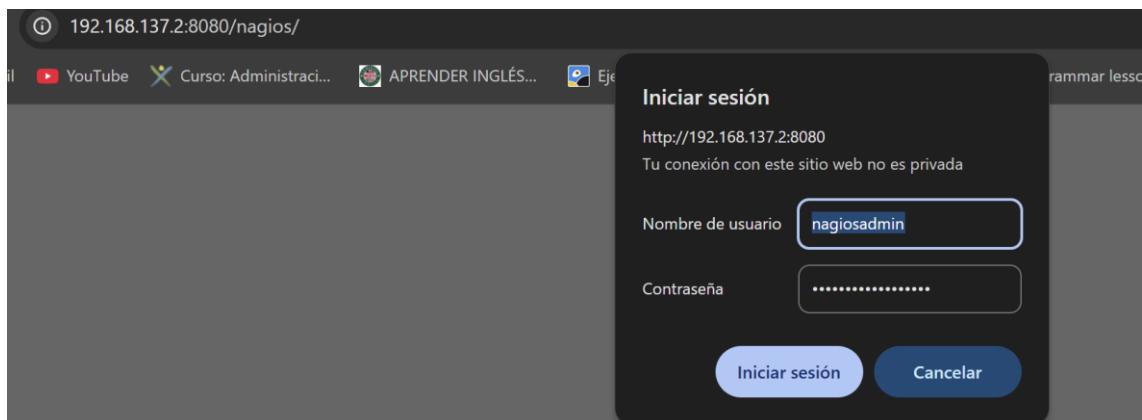
! Redirección de HTTP (puerto 80)

```
R2(config)# ip nat inside source static tcp 10.10.10.2 80 interface  
FastEthernet0/0 80
```

! Redirección de SSH (puerto 22)

```
R2(config)# ip nat inside source static tcp 10.10.10.2 22 interface  
FastEthernet0/0 22
```

Mediante la siguiente imagen se verifica la conexión desde mi pc fuera de la red aislada:



**Pantalla principal de Nagios Core:** Vista de bienvenida que muestra el estado del demonio, versión instalada y enlaces rápidos para documentación y noticias.

The screenshot shows the Nagios Core welcome page. On the left is a vertical navigation menu with sections like General, Current Status, Reports, and System. The 'Current Status' section is expanded, showing 'Tactical Overview' as the selected item. The main content area features the Nagios Core logo and version information ('Version 4.4.14'). It includes a 'Get Started' box with links to monitoring basics, a 'Latest News' box, a 'Don't Miss...' box, and a 'Quick Links' box with links to Nagios documentation and support. A 'Page Tour' link is visible on the right.

**Tactical Overview:** Resumen de problemas actuales con recuento de hosts y servicios en estados Down/Critical, y métricas de rendimiento y salud de la red.

This screenshot shows the 'Tactical Overview' page. The left sidebar has 'Tactical Overview' selected under 'Current Status'. The main area displays 'Network Outages' (0), 'Hosts' (1 Down, 0 Unreachable, 5 Up, 0 Pending), and 'Services' (5 Critical, 0 Warning, 0 Unknown, 17 Ok, 0 Pending). Below these are sections for 'Monitoring Features' (Flap Detection, Notifications, Event Handlers, Active Checks, Passive Checks) and 'Network Health' (Host Health and Service Health). A red arrow points from the 'Monitoring Features' section towards the bottom right of the page.

**Listado de Servicios por Host:** Tabla detallada de cada host con sus servicios (PING, SSH, HTTP, etc.), estado, último chequeo, duración y detalles de resultado.

## Interpretación global

Nagios Core cambia el color de los servicios automáticamente:

- **OK** – Servicio activo, dentro de los límites
- **WARNING** – Se superó el umbral de advertencia
- **Critical** – Se superó el umbral crítico o el servicio no responde
- **UNKNOWN** – Resultado no concluyente o error de configuración

| Host **       | Service **      | Status **    | Last Check **       | Duration **         | Attempt **    | Status Information  |
|---------------|-----------------|--------------|---------------------|---------------------|---------------|---|
| PC1           | PING            | OK           | 05-12-2025 12:27:28 | 0d 0h 26m 18s       | 1/4           | PING OK - Packet loss = 0%, RTA = 18.07 ms                            |
|               | SSH             | OK           | 05-12-2025 12:27:24 | 0d 2h 19m 19s       | 1/4           | SSH OK - OpenSSH_8.2p1 Ubuntu-Absubuntu0.4 (protocol 2.0)             |
| PC2           | PING            | OK           | 05-12-2025 12:27:10 | 0d 0h 26m 36s       | 1/4           | PING OK - Packet loss = 0%, RTA = 13.90 ms                            |
|               | SSH             | OK           | 05-12-2025 12:25:45 | 0d 2h 13m 4s        | 1/4           | SSH OK - OpenSSH_8.2p1 Ubuntu-Absubuntu0.4 (protocol 2.0)             |
| Router-R1     | PING            | OK           | 05-12-2025 12:23:20 | 0d 1h 10m 26s       | 1/4           | PING OK - Packet loss = 0%, RTA = 4.68 ms                             |
|               | SSH             | Critical     | 05-12-2025 12:27:38 | 0d 2h 13m 38s       | 4/4           | connect to address 10.10.10.1 and port 22: Connection refused         |
| Router-R2     | PING            | OK           | 05-12-2025 12:24:00 | 0d 2h 19m 5s        | 1/4           | PING OK - Packet loss = 0%, RTA = 22.54 ms                            |
|               | SSH             | Critical     | 05-12-2025 12:27:16 | 0d 2h 14m 42s       | 4/4           | connect to address 172.16.1.2 and port 22: Connection refused         |
| Server-Nagios | Current Load    | OK           | 05-12-2025 12:26:30 | 0d 2h 16m 21s       | 1/4           | OK - load average: 0.03, 0.01, 0.00                                   |
|               | Current Users   | OK           | 05-12-2025 12:22:51 | 0d 2h 16m 6s        | 1/4           | USERS OK - 1 users currently logged in                                |
|               | Disk Space      | OK           | 05-12-2025 12:24:13 | 0d 2h 18m 50s       | 1/4           | DISK OK - free space / 19087 MB (80.44% inode=92%):                   |
|               | HTTP            | OK           | 05-12-2025 12:26:13 | 0d 1h 6m 29s        | 1/4           | HTTP OK - HTTP/1.1 200 OK - 10945 bytes in 0.001 second response time |
|               | PING            | OK           | 05-12-2025 12:26:41 | 0d 2h 16m 7s        | 1/4           | PING OK - Packet loss = 0%, RTA = 0.10 ms                             |
|               | SSH             | OK           | 05-12-2025 12:23:06 | 0d 2h 15m 57s       | 1/4           | SSH OK - OpenSSH_8.9p1 Ubuntu-Subuntu0.10 (protocol 2.0)              |
|               | Total Processes | OK           | 05-12-2025 12:24:28 | 0d 2h 18m 38s       | 1/4           | PROCS OK: 180 processes   |
|               | Server-Syslog   | Current Load | OK                  | 05-12-2025 12:24:41 | 0d 2h 17m 16s | 1/4   |
|               | Current Users   | OK           | 05-12-2025 12:26:57 | 0d 2h 15m 54s       | 1/4           | USERS OK - 1 users currently logged in                                |
|               | Disk Space      | OK           | 05-12-2025 12:23:11 | 0d 2h 19m 40s       | 1/4           | DISK OK - free space / 19087 MB (80.44% inode=92%):                   |
|               | HTTP            | Critical     | 05-12-2025 12:24:41 | 0d 2h 14m 53s       | 4/4           | connect to address 10.10.11.2 and port 80: Connection refused         |
|               | PING            | OK           | 05-12-2025 12:24:29 | 0d 1h 8m 13s        | 1/4           | PING OK - Packet loss = 0%, RTA = 12.73 ms                            |
|               | SSH             | OK           | 05-12-2025 12:27:12 | 0d 1h 5m 31s        | 1/4           | SSH OK - OpenSSH_8.9p1 Ubuntu-Subuntu0.10 (protocol 2.0)              |
|               | Total Processes | OK           | 05-12-2025 12:23:32 | 0d 2h 17m 56s       | 1/4           | PROCS OK: 180 processes   |

Page four

**Service Status Details:** Detalle filtrado de servicios problemáticos, mostrando totales de estados Up/Down/Warning/Critical y entradas individuales con tiempos y mensajes.

**Nagios®**

**General**

- [Home](#)
- [Documentation](#)

**Current Status**

- [Tactical Overview](#)
- [Map \(Legacy\)](#)
- [Hosts](#)
- [Services](#)
- [Host Groups](#)
- [Summary Grid](#)
- [Service Groups](#)
- [Summary Grid](#)
- [Problems](#) (selected)
- [Services \(Unhandled\)](#)
- [Hosts \(Unhandled\)](#)
- [Network Outages](#)
- [Quick Search...](#)

Current Network Status  
Last Updated: Mon May 12 10:32:29 UTC 2025  
Updated every 90 seconds  
Nagios® Core™ 4.4.14 - www.nagios.org  
Logged in as nagiosadmin

[View History For All hosts](#)

[View Notifications For All Hosts](#)

[View Host Status Detail For All Hosts](#)

**Display Filters:**

- [Host Properties: All](#)
- [Host Properties: Any](#)
- [Service Status Types: All Problems](#)
- [Service Properties: Any](#)

Limit Results:  ▾

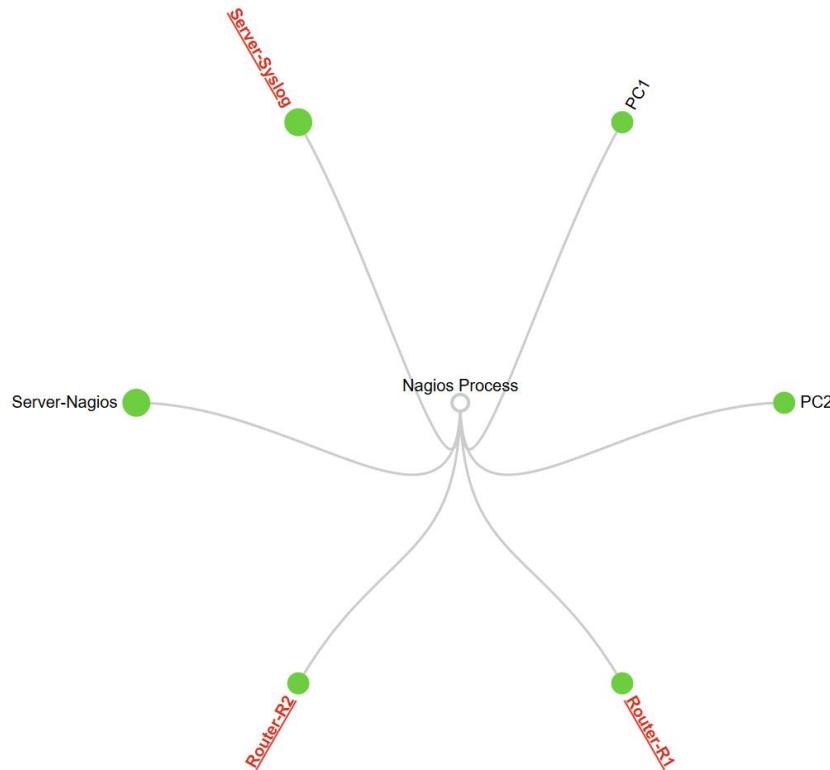
| Host **       | Service ** | Status ** | Last Check **       | Duration **   | Attempt ** | Status Information  |
|---------------|------------|-----------|---------------------|---------------|------------|---|
| PC2           | PING       | WARNING   | 05-12-2025 10:30:24 | 0d 0h 2m 5s   | 4/4        | PING WARNING - Packet loss = 28%, RTA = 17.08 ms              |
| Router-R1     | SSH        | Critical  | 05-12-2025 10:29:04 | 0d 0h 18m 25s | 4/4        | connect to address 10.10.10.1 and port 22: Connection refused |
| Router-R2     | SSH        | Critical  | 05-12-2025 10:28:00 | 0d 0h 19m 29s | 4/4        | connect to address 172.16.1.2 and port 22: Connection refused |
| Server-Syslog | HTTP       | Critical  | 05-12-2025 10:27:59 | 0d 0h 19m 40s | 4/4        | Critical - Socket timeout                                     |
|               | PING       | Critical  | 05-12-2025 10:30:42 | 0d 0h 18m 18s | 4/4        | PING CRITICAL - Packet loss = 100%                            |
|               | SSH        | Critical  | 05-12-2025 10:30:43 | 0d 0h 19m 56s | 4/4        | Critical - Socket timeout                                     |

Results 1 - 6 of 6 Matching Services

**Alert History:** Registro cronológico de alertas de hosts y servicios, incluyendo eventos de flapping, warnings y criticals extraídos del log de Nagios.

The screenshot shows the Nagios Alert History interface. On the left, there's a sidebar with various navigation links: Home, Documentation, Current Status, Tactical Overview Map (Legacy), Hosts, Services, Host Groups, Service Groups, Problems, Reports, System, and Configuration. The 'Current Status' section is active. The main content area has three tabs: 'Alert History' (selected), 'All Hosts and Services', and 'Log File Navigation'. The 'Alert History' tab shows a list of alerts with timestamps and descriptions. A date range selector 'Latest Archive' is at the top, followed by 'File: /usr/local/nagios/var/nagios.log'. On the right, there are 'State type options' dropdowns and checkboxes for filtering alerts based on severity (All state types, All alerts, Hide Flapping Alerts, Hide Downtime Alerts, Hide Process Messages, Older Entries First) and an 'Update' button.

**Network Map (Status Map):** Diagrama gráfico de la topología de monitorización, con el proceso Nagios en el centro conectando a cada nodo supervisado.



### 3.17 Extracción de los datos del Servidor Nagios

Para extraer los datos de Nagios primero debes asegurarte de que R2 permite el acceso SSH a la red de Nagios y que realiza **el port-forwarding correctamente**.

Una vez dentro, los ficheros de estado, colas y logs de Nagios residen en **/usr/local/nagios/var**:

```
alumno@nagios:/usr/local/nagios/var$ tree
.
├── archives
├── nagios.log
├── nagios.tmp08dkLy
├── nagios.tmp5HrQIV
├── nagios.tmpTUxZuA
├── objects.cache
├── retention.dat
└── rw
    ├── nagios.cmd
    └── nagios.qh
└── spool
    └── checkresults
    └── status.dat

4 directories, 9 files
alumno@nagios:/usr/local/nagios/var$
```

**Empleamos cmd scp para copiar los datos:**

Desde mi equipo, se utiliza scp especificando el puerto mapeado (2222) para descargar los ficheros

En Windows (PowerShell o CMD):

```
scp -P 2222 alumno@192.168.137.2:"/usr/local/nagios/var/nagios-
4.4.14.tar.gz" C:\Users\carla\Downloads\
```

```
alumno@192.168.137.2's password:
scp: Connection closed

C:\Users\carla\Downloads>scp -P 2222 "C:\Users\carla\Downloads\nagios-4.4.14.tar.gz" alumno@192.168.137.2:/home/alumno/
alumno@192.168.137.2's password:
nagios-4.4.14.tar.gz                                         100%   11MB   9.0KB/s   20:38
```

*Imagen 20 Descarga del directorio var de Nagios*

### 3.18 Marco Teórico de Power BI

Power BI nace de la evolución de las herramientas de Microsoft en el ámbito de la inteligencia de negocio, buscando democratizar el acceso al análisis de datos. Su arquitectura modular permite escalar desde proyectos personales hasta soluciones corporativas con grandes volúmenes de información.

En la capa de **Power Query**, el lenguaje M facilita la limpieza y transformación de datos mediante funciones predefinidas y la grabación de pasos, asegurando trazabilidad y repetibilidad de los procesos ETL. Una vez los datos están listos, **Power Pivot** aprovecha el motor xVelocity in-memory para almacenar grandes tablas en memoria comprimida y aplicar el lenguaje DAX (Data Analysis Expressions), que ofrece potentes funciones para cálculos dinámicos, jerarquías y relaciones entre tablas.

La fase de **modelado** no solo consiste en unir tablas, sino también en diseñar métricas coherentes (medidas y columnas calculadas) y optimizar el rendimiento definiendo correctamente formatos de datos, cardinalidades y configuraciones de filtrado.

Por último, **Power View** y el servicio en la nube de Power BI permiten construir informes y dashboards interactivos, con filtros, segmentaciones y herramientas de exploración que habilitan el análisis self-service. Gracias a su integración con otras soluciones de Microsoft (Excel, Azure, Dynamics 365) y conectores a multitud de orígenes, Power BI se ha consolidado como una de las plataformas BI más versátiles y adoptadas en la empresa moderna.

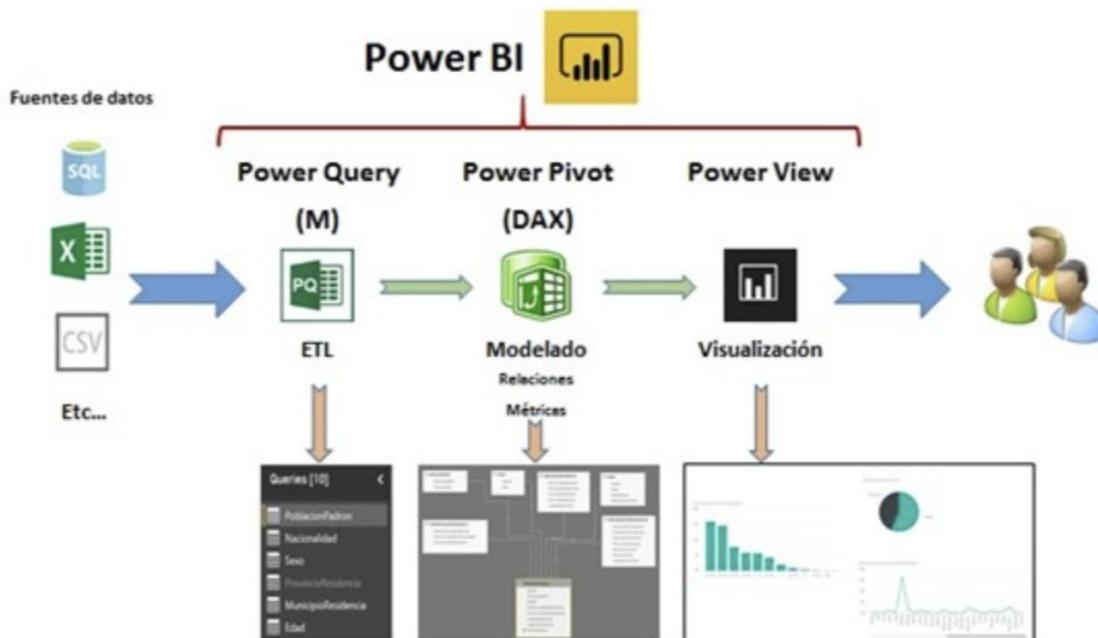


Imagen 21 Esquema de funcionamiento de Power Bi

### 3.19 Diseño e Implementación del Sistema de Análisis

Para alimentar Power BI con los datos de Nagios y Syslog, se ha optado por **volcar la información a una base de datos MySQL alojada en un servidor local**.

En primer lugar, se diseñó un **modelo relacional** que recoge las principales fuentes de datos: las alertas generadas por los servidores (nagios\_alerts), los eventos del registro de Nagios (nagios\_eventlog), los mensajes de syslog (syslog\_events), así como las tablas maestras de hosts, servicios, interfaces y tipos de interfaz.

Cada tabla contiene una **clave primaria única y las relaciones de integridad** referencial se han definido con claves foráneas:

por ejemplo, nagios\_alerts.host\_id y nagios\_alerts.service\_id apuntan a hosts.id y services.id respectivamente, mientras que host\_interfaces.host\_id se enlaza con hosts.id y host\_interfaces.interface\_id con interface\_types.id.

Las **cardinalidades (1:N, N:M)** se corresponden con la naturaleza de las entidades, garantizando la correcta asociación de múltiples alertas y eventos a un mismo host o servicio.

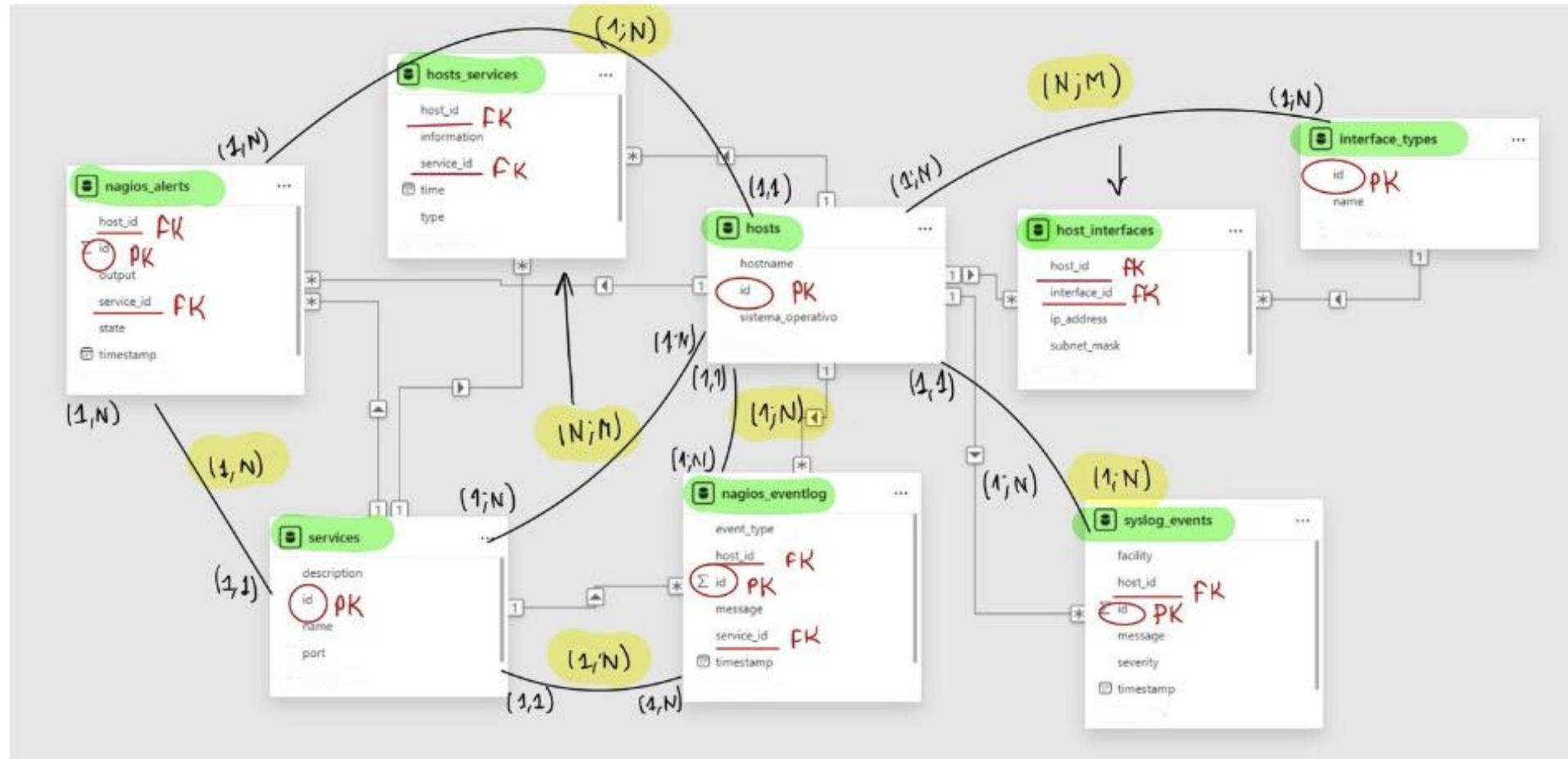


Imagen 22 Modelo relacional de la base de datos

Una vez establecido el esquema, se configuró un entorno MySQL sobre XAMPP: tras arrancar el servidor MySQL en el puerto 3370, se procedió a asegurar el acceso de administrador con el comando:

**ALTER USER 'root'@'localhost' IDENTIFIED BY 'TuNuevaContra123!';  
FLUSH PRIVILEGES;**

```
MariaDB [(none)]> ALTER USER 'root'@'localhost' IDENTIFIED BY 'TuNuevaContra123!';
Query OK, 0 rows affected (0.008 sec)

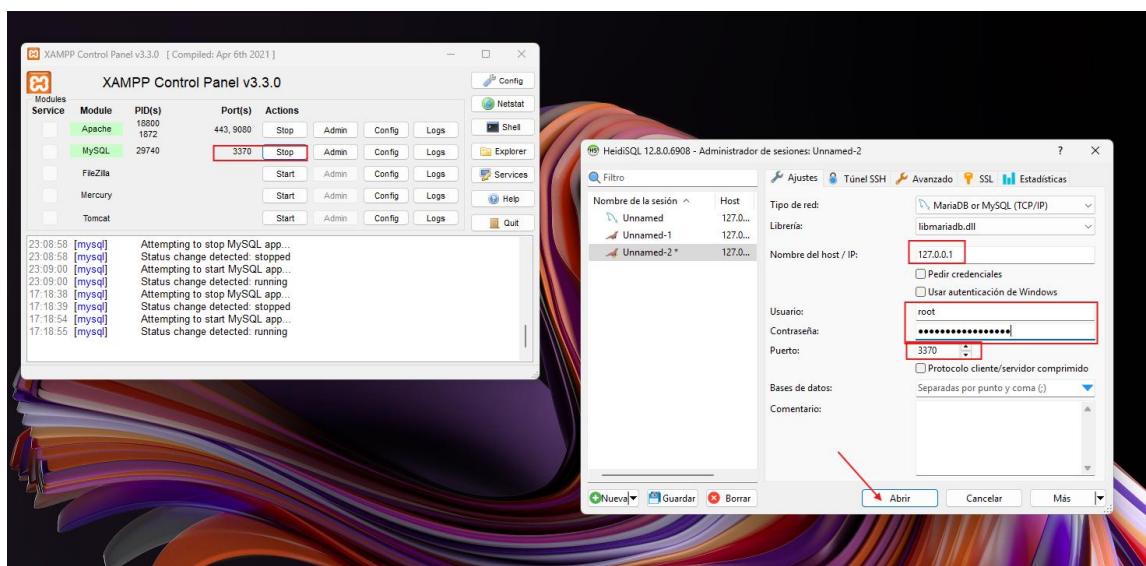
MariaDB [(none)]> FLUSH PRIVILEGES;
ERROR 1030 (HY000): Got error 176 "Read page with wrong checksum" from storage engine Aria
MariaDB [(none)]> ALTER USER 'root'@'localhost' IDENTIFIED BY 'TuNuevaContra123!';
Query OK, 0 rows affected (0.008 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
ERROR 1030 (HY000): Got error 176 "Read page with wrong checksum" from storage engine Aria
MariaDB [(none)]> SELECT Host, User, authentication_string
    ->     FROM mysql.user
    ->     WHERE User = 'root';
+-----+-----+
| Host | User | authentication_string |
+-----+-----+
| localhost | root | *CA4525D7EE393B807F2BC84386BBEA538124C1A3 |
| 127.0.0.1 | root |                               |
| ::1 | root |                               |
+-----+-----+
3 rows in set (0.057 sec)

MariaDB [(none)]> EXIT;
Bye

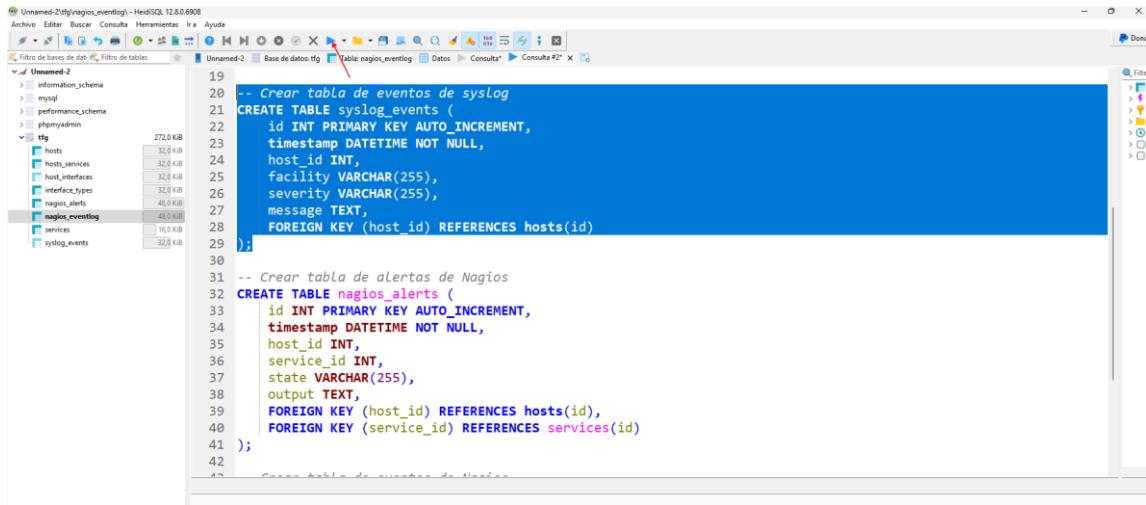
C:\Users\carla\xamp\mysql\bin>
```

*Imagen 23 Configuración de la contraseña del servidor local*



*Imagen 24 Acceso mediante Heidi al entorno MySQL*

A continuación, mediante el cliente gráfico HeidiSQL se creó la base de datos tfg y se ejecutaron los scripts SQL de creación de tablas, generando un total de siete tablas InnoDB con sus correspondientes relaciones.

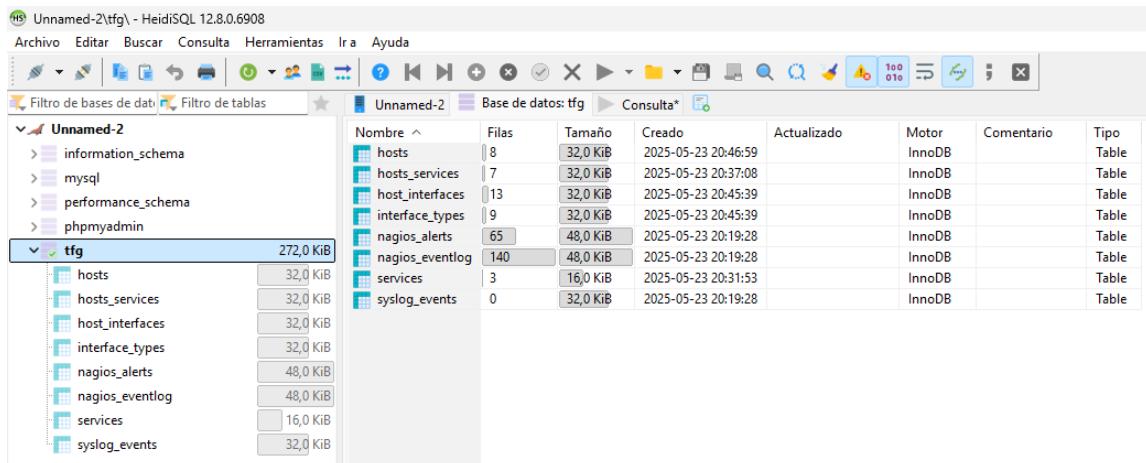


```

19
20 -- Crear tabla de eventos de syslog
21 CREATE TABLE syslog_events (
22     id INT PRIMARY KEY AUTO_INCREMENT,
23     timestamp DATETIME NOT NULL,
24     host_id INT,
25     facility VARCHAR(255),
26     severity VARCHAR(255),
27     message TEXT,
28     FOREIGN KEY (host_id) REFERENCES hosts(id)
29 );
30
31 -- Crear tabla de alertas de Nagios
32 CREATE TABLE nagios_alerts (
33     id INT PRIMARY KEY AUTO_INCREMENT,
34     timestamp DATETIME NOT NULL,
35     host_id INT,
36     service_id INT,
37     state VARCHAR(255),
38     output TEXT,
39     FOREIGN KEY (host_id) REFERENCES hosts(id),
40     FOREIGN KEY (service_id) REFERENCES services(id)
41 );
42

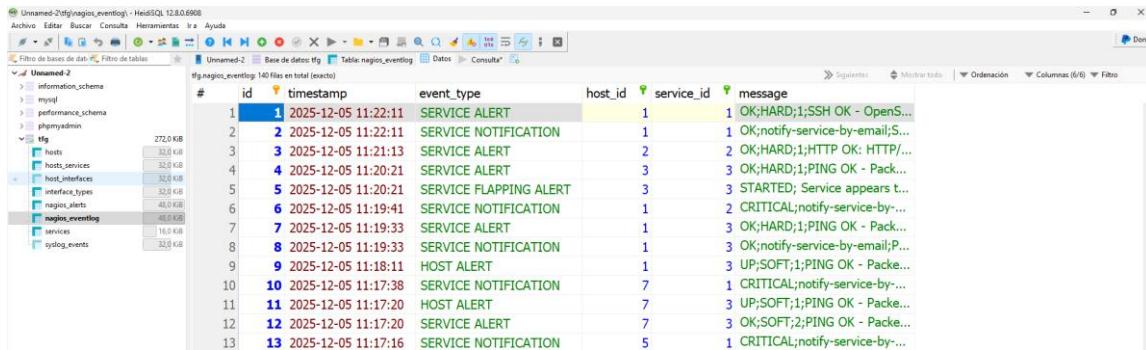
```

Imagen 25 Estructura SQL



| Nombre          | Filas | Tamaño   | Creado              | Actualizado | Motor  | Comentario | Tipo  |
|-----------------|-------|----------|---------------------|-------------|--------|------------|-------|
| hosts           | 8     | 32,0 KiB | 2025-05-23 20:46:59 |             | InnoDB |            | Table |
| hosts_services  | 7     | 32,0 KiB | 2025-05-23 20:37:08 |             | InnoDB |            | Table |
| host_interfaces | 13    | 32,0 KiB | 2025-05-23 20:45:39 |             | InnoDB |            | Table |
| interface_types | 9     | 32,0 KiB | 2025-05-23 20:45:39 |             | InnoDB |            | Table |
| nagios_alerts   | 65    | 48,0 KiB | 2025-05-23 20:19:28 |             | InnoDB |            | Table |
| nagios_eventlog | 140   | 48,0 KiB | 2025-05-23 20:19:28 |             | InnoDB |            | Table |
| services        | 3     | 16,0 KiB | 2025-05-23 20:31:53 |             | InnoDB |            | Table |
| syslog_events   | 0     | 32,0 KiB | 2025-05-23 20:19:28 |             | InnoDB |            | Table |

Imagen 26 Base de datos tfg



| #  | id | timestamp           | event_type             | host_id | service_id | message                           |
|----|----|---------------------|------------------------|---------|------------|-----------------------------------|
| 1  | 1  | 2025-12-05 11:22:11 | SERVICE ALERT          | 1       |            | 1 OK;HARD;1;SSH OK - OpenS...     |
| 2  | 2  | 2025-12-05 11:22:11 | SERVICE NOTIFICATION   | 1       |            | 1 OK;notify-service-by-email;S... |
| 3  | 3  | 2025-12-05 11:21:13 | SERVICE ALERT          | 2       |            | 2 OK;HARD;1;HTTP OK: HTTP/...     |
| 4  | 4  | 2025-12-05 11:20:21 | SERVICE ALERT          | 3       |            | 3 OK;HARD;1;PING OK - Pack...     |
| 5  | 5  | 2025-12-05 11:20:21 | SERVICE FLAPPING ALERT | 3       |            | 3 STARTED; Service appears t...   |
| 6  | 6  | 2025-12-05 11:19:41 | SERVICE NOTIFICATION   | 1       |            | 2 CRITICAL;notify-service-by...   |
| 7  | 7  | 2025-12-05 11:19:33 | SERVICE ALERT          | 1       |            | 3 OK;HARD;1;PING OK - Pack...     |
| 8  | 8  | 2025-12-05 11:19:33 | SERVICE NOTIFICATION   | 1       |            | 3 OK;notify-service-by-email;P... |
| 9  | 9  | 2025-12-05 11:18:11 | HOST ALERT             | 1       |            | 3 UP;SOFT;1;PING OK - Pack...     |
| 10 | 10 | 2025-12-05 11:17:38 | SERVICE NOTIFICATION   | 7       |            | 1 CRITICAL;notify-service-by...   |
| 11 | 11 | 2025-12-05 11:17:20 | HOST ALERT             | 7       |            | 3 UP;SOFT;1;PING OK - Pack...     |
| 12 | 12 | 2025-12-05 11:17:20 | SERVICE ALERT          | 7       |            | 3 OK;SOFT;2;PING OK - Pack...     |
| 13 | 13 | 2025-12-05 11:17:16 | SERVICE NOTIFICATION   | 5       |            | 1 CRITICAL;notify-service-by...   |

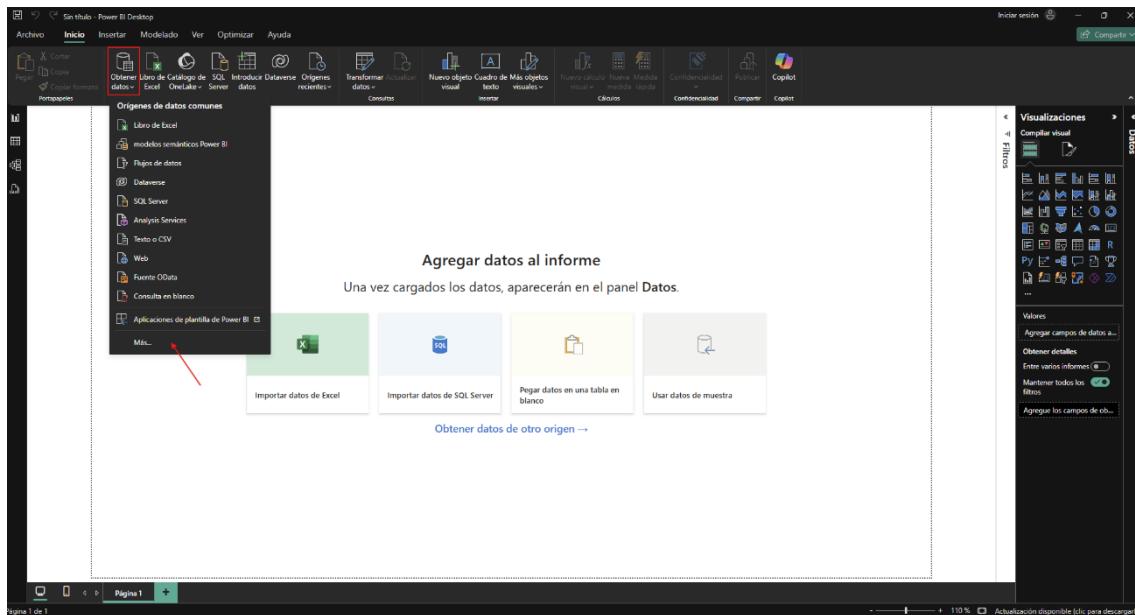
Imagen 27 Datos de tfg

### 3.20 Modelado de datos

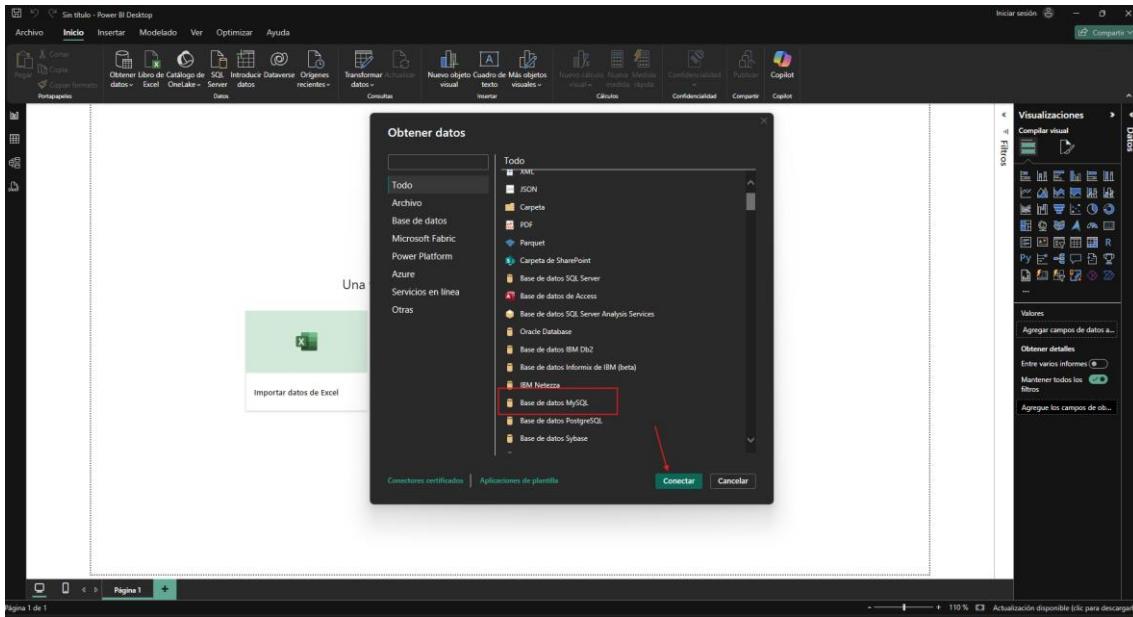
El modelado de datos es un **proceso fundamental** dentro del ciclo de vida de cualquier sistema de información, pues permite representar de forma estructurada y comprensible el flujo y la organización de los datos que se almacenan, procesan y comparten.

En esta sección se aborda **cómo diseñar modelos de datos** que reflejen fielmente los requisitos del negocio, garanticen la integridad de la información y faciliten tanto el acceso eficiente como la óptima escalabilidad de la solución.

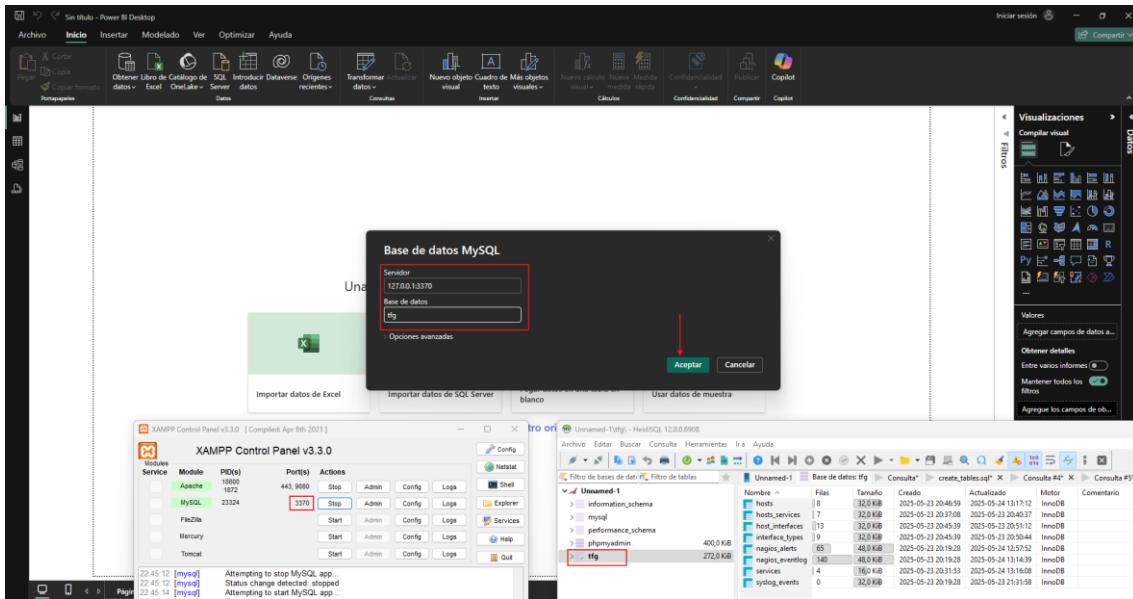
**Se abre** Power BI Desktop y, en la pestaña **Inicio**, se hace clic en **Obtener datos** → **Más....**



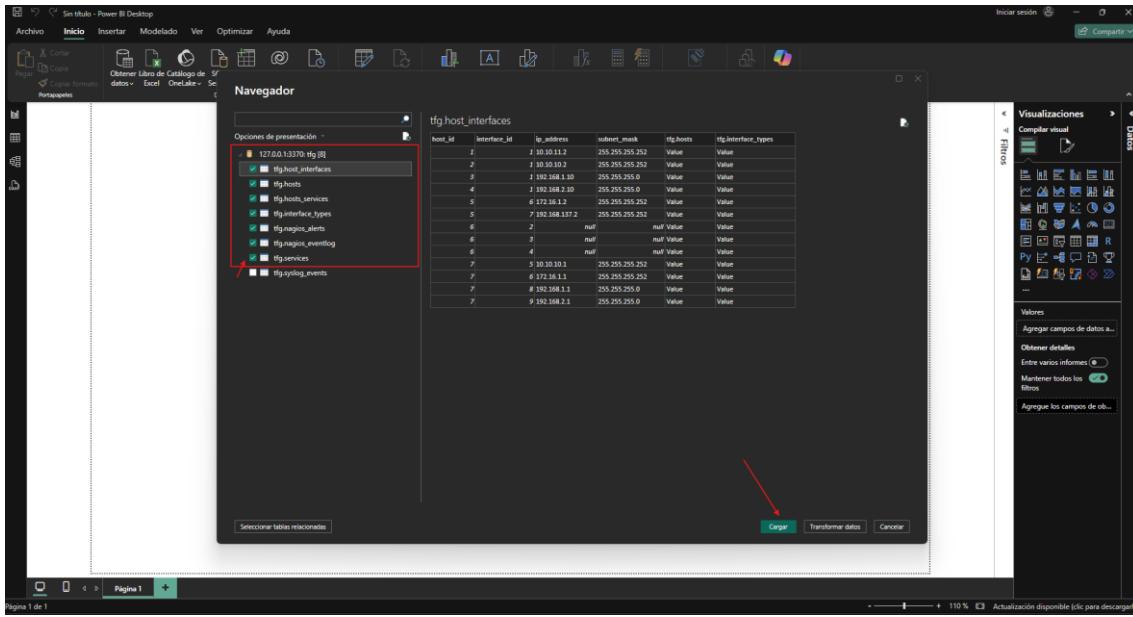
**Se selecciona en el cuadro Obtener datos la opción Base de datos MySQL y se pulsa Conectar.**



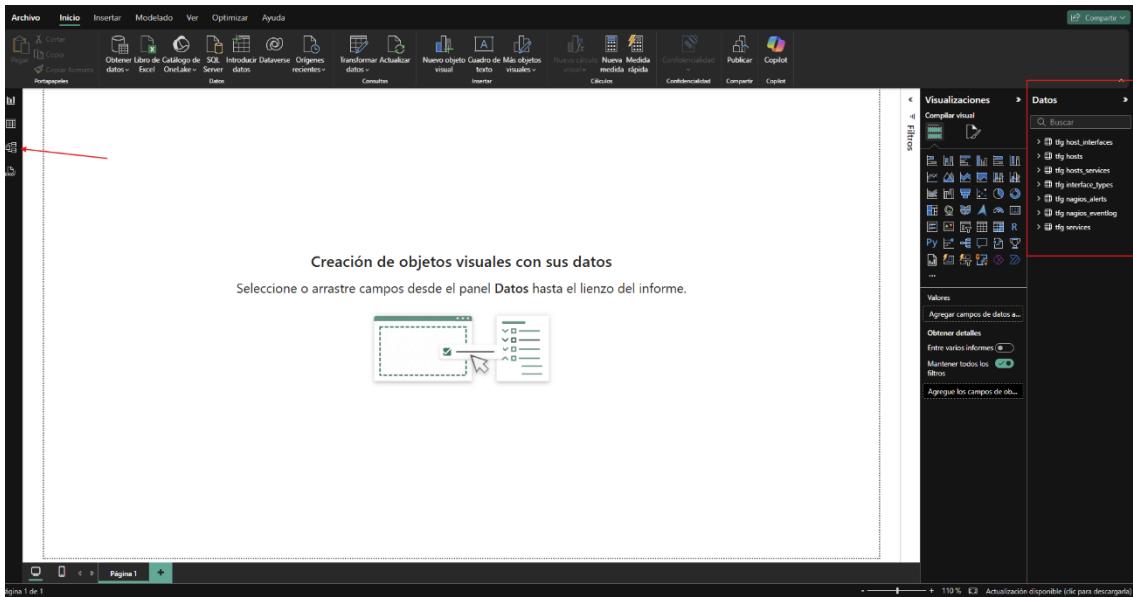
**Se introduce** en el diálogo **Servidor** la dirección 127.0.0.1:3370 y en **Base de datos** el nombre tfg, luego se pulsa **Aceptar**.



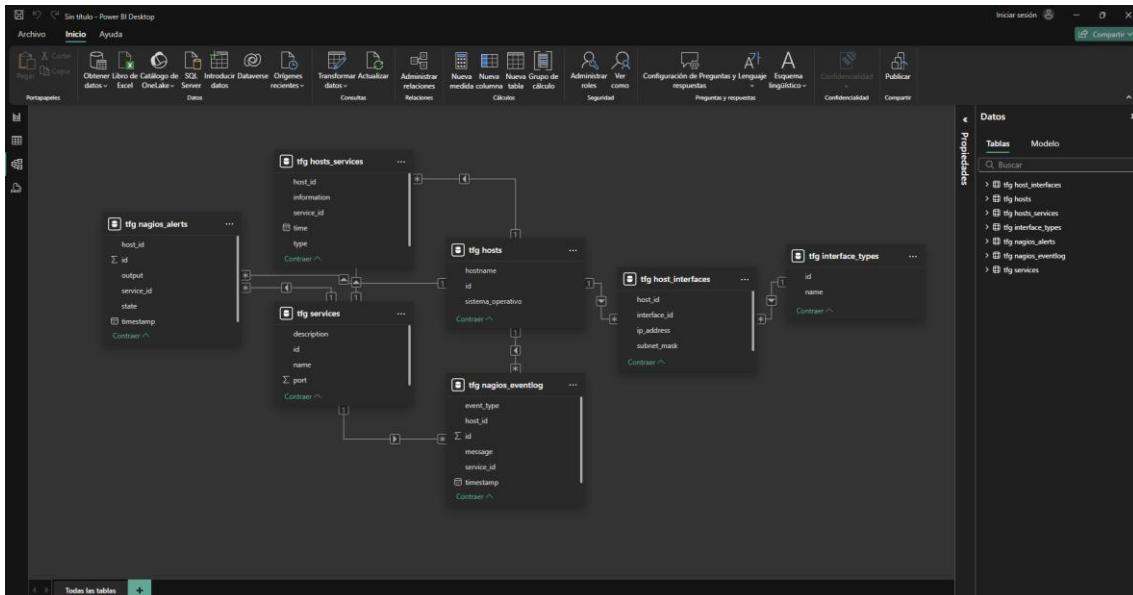
**Se espera** a que Power BI muestre el **Navegador** con las tablas de tfg; allí se marcan **tfg.hosts**, **tfg.services**, **tfg.host\_interfaces**, **tfg.interface\_types**, **tfg.nagios\_alerts** y **tfg.nagios\_eventlog**, y se pulsa **Cargar**.



**Se comprueba** en el panel **Campos** que todas las tablas se han importado correctamente bajo el prefijo tfg.



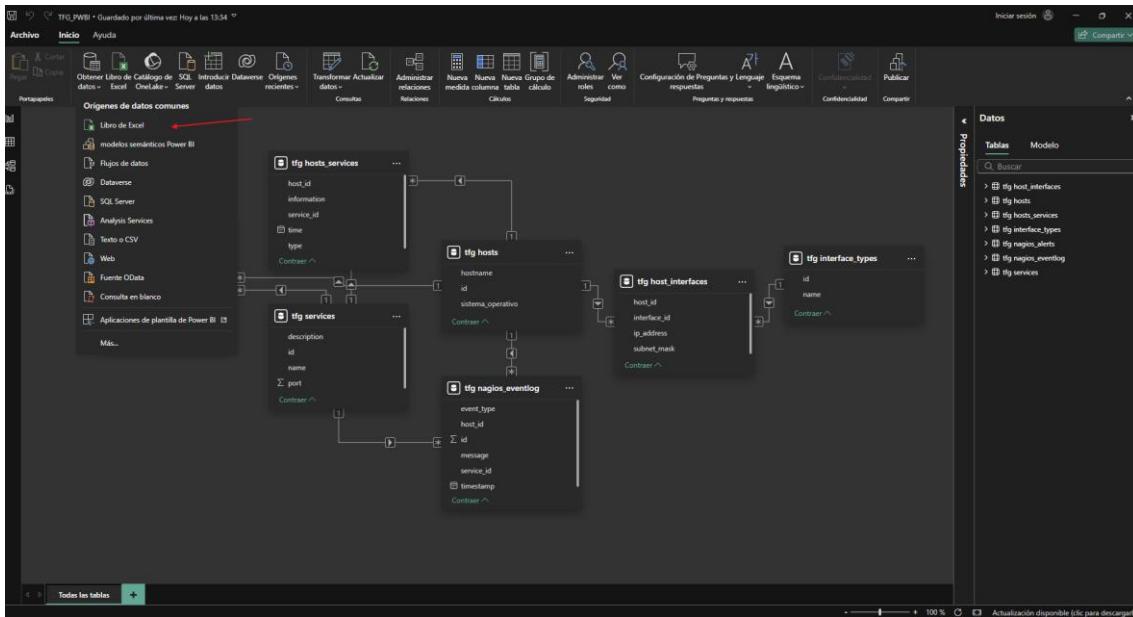
**Se cambia** a la vista **Modelo** (ícono de diagrama) y se observa cómo Power BI ha trazado automáticamente las relaciones PK–FK entre las tablas importadas.



Como falta tabla debido a tu gran tamaño la hacemos manualmente por Excel,

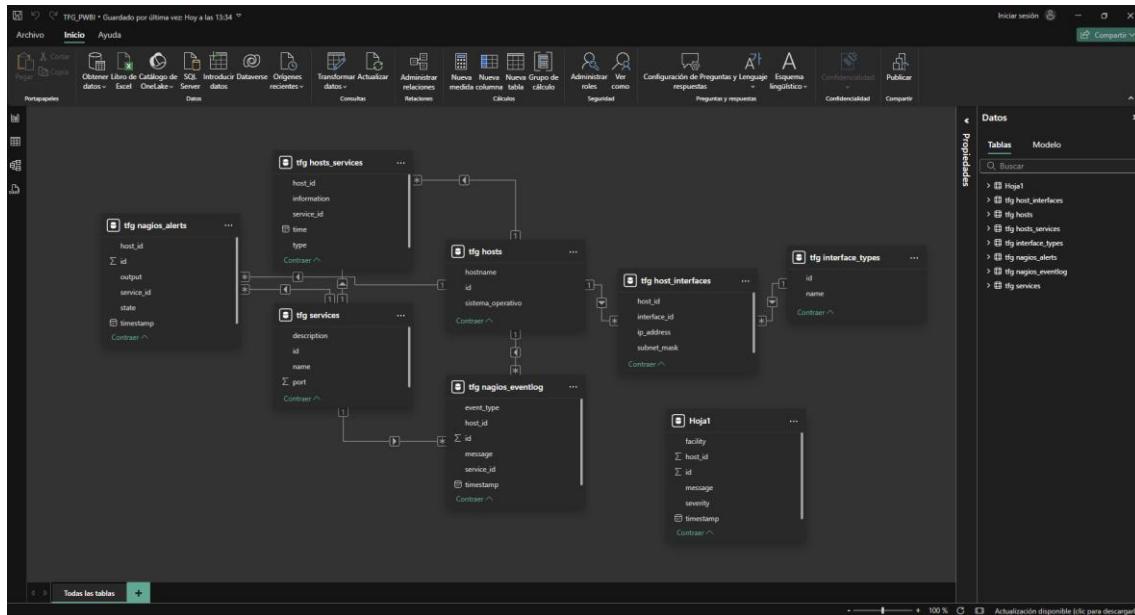
| A  | B  | C               | D       | E        | F        | G  | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W |  |
|----|----|-----------------|---------|----------|----------|--|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|
| 1  | id | timestamp       | host_id | facility | severity | message  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 2  |    | 1 2025-04-27 1  | 3       | cron     | info     | (CRON) INFO: profile: fd=3   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 3  |    | 2 2025-04-27 1  | 3       | cron     | info     | (CRON) INFO: (pidfile fd=0) Skipping @reboot jobs -- not system startup  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 4  |    | 3 2025-04-27 1  | 3       | cron     | info     | (CRON) INFO: (pidfile fd=3)  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 5  |    | 4 2025-04-27 1  | 3       | cron     | info     | (CRON) INFO: Skipping @reboot jobs -- not system startup   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 6  |    | 5 2025-04-27 1  | 3       | daemon   | info     | rsyslog.service: Successed.  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 7  |    | 6 2025-04-27 1  | 3       | daemon   | info     | Stopped System Logging Service.  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 8  |    | 7 2025-04-27 1  | 3       | daemon   | info     | Starting System Logging Service...   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 9  |    | 8 2025-04-27 1  | 3       | daemon   | info     | Started System Logging Service.  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 10 |    | 9 2025-04-27 1  | 3       | daemon   | info     | levelerror msg="data were not delivered successfully to metrics server, retrying in 1800s"   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 11 |    | 10 2025-04-27 1 | 3       | daemon   | info     | levelerror msg="data were not delivered successfully to metrics server, retrying in 1800s"   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 12 |    | 11 2025-04-27 1 | 3       | daemon   | info     | Started Run amazon jobs.   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 13 |    | 12 2025-04-27 1 | 3       | daemon   | info     | amazon.jobs: Successed.  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 14 |    | 13 2025-04-27 1 | 3       | daemon   | info     | [system] Activating via systemd: service name="net.reactivated.fprint" unit="fprintd.service" requested by "1.99" (uid=1000 pid=1637 comm="/usr/bin/gnome-shell" label="unconfined") |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 15 |    | 14 2025-04-27 1 | 3       | daemon   | info     | Starting Fingerprint Authentication Daemon...  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 16 |    | 15 2025-04-27 1 | 3       | daemon   | info     | [system] Successfully activated service "net.reactivated.fprint"   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 17 |    | 16 2025-04-27 1 | 3       | daemon   | info     | Started Fingerprint Authentication Daemon.   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 18 |    | 17 2025-04-27 1 | 3       | daemon   | info     | [info] [17407010] agent registered agent[12459500451211..1.99/org.gnome.Shell.NetworkAgent/2000]: agent registered.  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 19 |    | 18 2025-04-27 1 | 3       | daemon   | info     | [session uid=1000 pid=1375] Activating service name="org.freedesktop.FileManager1" requested by "1.42" (uid=1000 pid=1637 comm="/usr/bin/gnome-shell" label="unconfined")            |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 20 |    | 19 2025-04-27 1 | 3       | daemon   | info     | [session uid=1000 pid=1375] Activating service name="org.gnome.Nautilus" requested by "1.42" (uid=1000 pid=1637 comm="/usr/bin/gnome-shell" label="unconfined")                      |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 21 |    | 20 2025-04-27 1 | 3       | daemon   | info     | [session uid=1000 pid=1375] Successfully activated service "org.gnome.Nautilus"  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 22 |    | 21 2025-04-27 1 | 3       | daemon   | info     | Window manager: Overwriting existing binding of keyym 33 with keyym 33 (keycode 1).  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 23 |    | 22 2025-04-27 1 | 3       | daemon   | info     | Window manager: Overwriting existing binding of keyym 35 with keyym 35 (keycode 2).  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 24 |    | 23 2025-04-27 1 | 3       | daemon   | info     | Window manager: Overwriting existing binding of keyym 34 with keyym 34 (keycode d).  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 25 |    | 24 2025-04-27 1 | 3       | daemon   | info     | Window manager: Overwriting existing binding of keyym 36 with keyym 36 (keycode f).  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 26 |    | 25 2025-04-27 1 | 3       | daemon   | info     | Window manager: Overwriting existing binding of keyym 38 with keyym 38 (keycode 11).   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 27 |    | 26 2025-04-27 1 | 3       | daemon   | info     | Window manager: Overwriting existing binding of keyym 39 with keyym 39 (keycode 12).   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 28 |    | 27 2025-04-27 1 | 3       | daemon   | info     | Window manager: Overwriting existing binding of keyym 37 with keyym 37 (keycode 13).   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 29 |    | 28 2025-04-27 1 | 3       | daemon   | info     | Window manager: Overwriting existing binding of keyym 30 with keyym 30 (keycode 12).   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 30 |    | 29 2025-04-27 1 | 3       | daemon   | info     | Window manager: Overwriting existing binding of keyym 36 with keyym 36 (keycode f).  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 31 |    | 30 2025-04-27 1 | 3       | daemon   | info     | Window manager: Overwriting existing binding of keyym 37 with keyym 37 (keycode 10).   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 32 |    | 31 2025-04-27 1 | 3       | daemon   | info     | fprintd.service: Successed.  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 33 |    | 32 2025-04-27 1 | 3       | daemon   | info     | Reloading...   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 34 |    | 33 2025-04-27 1 | 3       | daemon   | info     | Starting process error reports when automatic reporting is enabled...  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 35 |    | 34 2025-04-27 1 | 3       | daemon   | info     | Reloading...   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 36 |    | 35 2025-04-27 1 | 3       | daemon   | info     | skipping, not a crash  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 37 |    | 36 2025-04-27 1 | 3       | daemon   | info     | message repeated 7 times: [ skipping, not a crash ]  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 38 |    | 37 2025-04-27 1 | 3       | daemon   | info     | /var/crash/_utl_low_systemd_systemd-logind.0.crash already marked for upload, skipping   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |
| 39 |    | 38 2025-04-27 1 | 3       | daemon   | info     | /var/crash/_utl_low_systemd_systemd-timesyncd.0.crash already marked for upload, skipping  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |

Se vuelve a Obtener datos → Más..., pero esta vez se elige Excel y se pulsa Conectar.

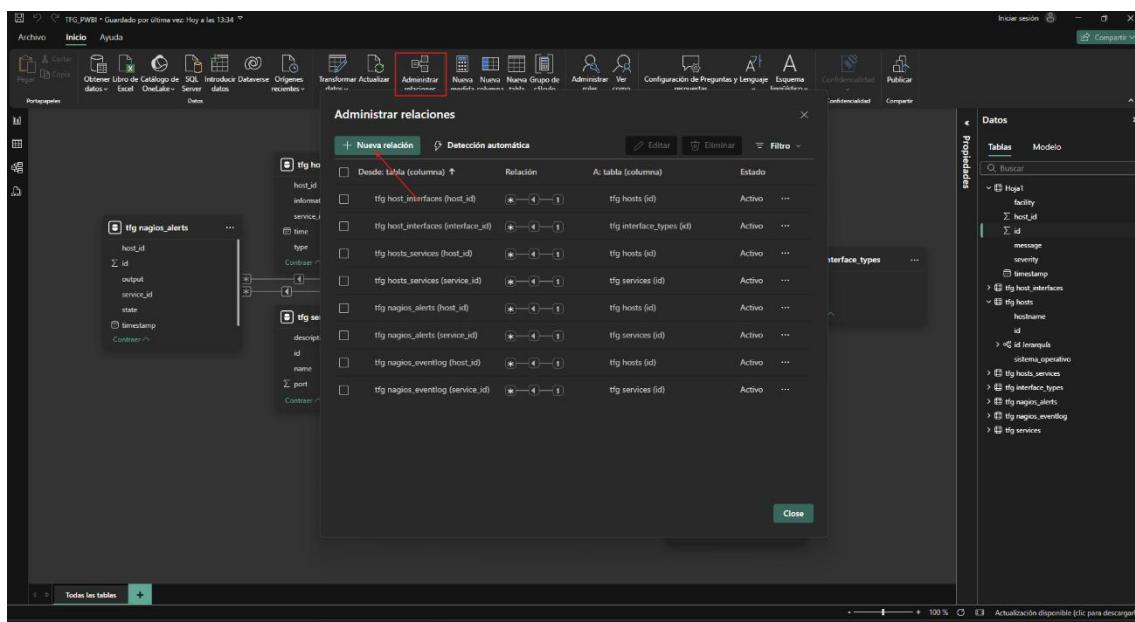


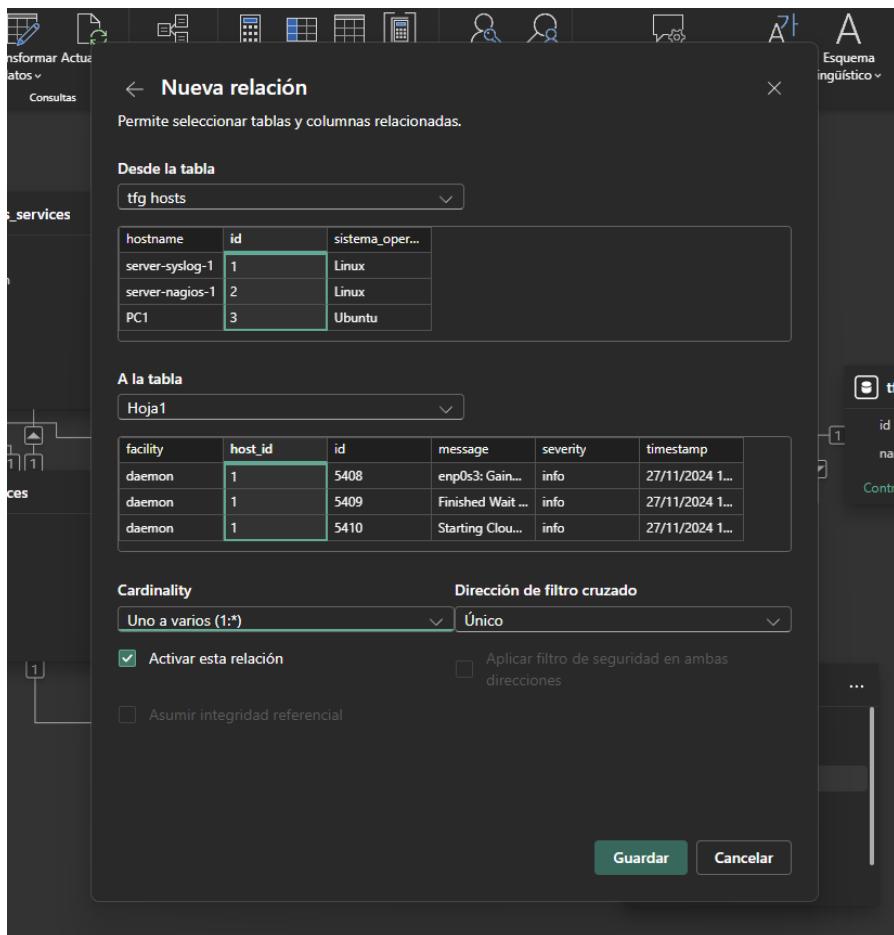
**Se localiza** el fichero exportado de syslog (.xlsx) y se selecciona la hoja que contiene las columnas timestamp, host\_id, facility, severity, message; a continuación, se pulsa **Cargar**.

**Se verifica** en el panel **Campos** la nueva tabla (por ejemplo, **Hoja1**) con los datos de syslog.



Se regresa a la vista **Modelo** y, arrastrando, se une Hoja1[host\_id] con tlg.hosts[id] para establecer la relación entre mensajes de syslog y hosts.





**Se guarda** el proyecto de Power BI, quedando así disponible el modelo completo para crear informes y visualizaciones.

### 3.21 Creación de medidas y KPIs en DAX y Desarrollo de Dashboards y Reportes

La fuente **syslog** (servidores Linux) se almacena en la tabla **syslog\_events** dentro del modelo de datos. Esta tabla recoge todos los **eventos de sistema y de autenticación**, permitiendo analizar tanto los sucesos generales del sistema como los intentos de acceso, cambios de sesión y otros registros críticos de seguridad.

La plataforma de **Nagios** (monitorización de infraestructura) se integra en dos tablas del modelo: **nagios\_alerts** y **nagios\_eventlog**. En **nagios\_alerts** se registran las **alertas de estado** de hosts y servicios (por ejemplo, transiciones de OK a WARNING o CRITICAL), mientras que **nagios\_eventlog** documenta la **evolución de esos estados** —desde los eventos SOFT iniciales hasta la confirmación HARD—, lo que facilita el seguimiento detallado de la salud de la infraestructura a lo largo del tiempo.

Los **KPIs** (Key Performance Indicators) que calculas con DAX sirven para responder a tres preguntas básicas de operación:

1. **¿Está pasando algo?** → cuántos incidentes y de qué tipo.
2. **¿Qué gravedad tiene?** → severidad, porcentaje de caída, escaladas.
3. **¿Estamos reaccionando bien?** → SLA, falsos positivos, latencia, notificaciones.

En Power BI distinguimos dos formas de extender el modelo de datos con lógica propia: **medidas** y **columnas calculadas**.

- **Medida (Measure)**

Una medida es un cálculo dinámico que se evalúa en función del contexto de filtro de los visuales (la selección de fechas, categorías, regiones...).

Se crean desde **Vista Modelado** → **Nueva medida**, pegando la fórmula DAX en la barra de expresiones. Al vivir en el contexto de filtro, suelen usar funciones como CALCULATE, FILTER o variables (VAR) para controlar precisamente qué datos participan en el resultado.

- **Columna calculada (Calculated Column)**

Una columna calculada añade un nuevo campo a la tabla, evaluándose **fila a fila** en el momento de procesar el modelo.

Se definen en **Vista Modelado** → **Nueva columna** y resultan útiles cuando cada registro necesita una clasificación o atributo derivado (por ejemplo, AuthOutcome, state\_type o EsFalsoPositivo).

Los **29 KPIs** cubren desde disponibilidad (SLA) hasta la calidad de las alertas:

1. **Disponibilidad:** %HostsUP, Latencia PING.
2. **Severidad:** Incidentes Críticos, Eventos\_WARNING.
3. **Seguridad:** AuthFails, %AuthFail.
4. **Calidad de monitorización:** Soft→Hard, %FalsosPositivos.
5. **Eficacia operativa:** Incidentes fuera de SLA, Notificaciones 24×7.

### 3.21.1 Incidencias del sistema (tabla syslog\_events)

| KPI                              | Tipo         | Fórmula (DAX)                                 | Explicación   |
|----------------------------------|--------------|---|---|
| <b>Total de Incidentes</b>       | Medida       | Total Incidents = COUNT( syslog_events[id] )  | «¿Cuántas filas de syslog se consideran “incidente”?» Se suele mostrar en una tarjeta.      |
| <b>Eventos por severidad</b>     | Medida       | COUNT ( syslog_events[id] )                   | Usa segmentaciones para ver cuántos <i>error</i> , <i>warning</i> ...                       |
| <b>AuthOutcome</b>               | Columna      | (ver código en el anexo)                      | Clasifica cada evento auth en <b>Fail / Success / Other</b> usando SWITCH + CONTAINSSTRING. |
| <b>AuthFails<br/>AuthSuccess</b> | Medidas      | (ver código en el anexo)                      | Filtran la tabla por el valor de AuthOutcome.   |
| <b>% Auth Fail</b>               | Medida extra | DIVIDE([AuthFails],[AuthFails]+[AuthSuccess]) | Ratio de intentos fallidos; un disparador de alerta de seguridad.                           |

### 3.21.2 Alertas de infraestructura (tabla nagios\_alerts)

| KPI                             | Fórmula abreviada                        | ¿Qué indica?   |
|---------------------------------|--|--|
| %HostsUP                        | DIVIDE(EventosOK, EventosOK+EventosNoOK) | Disponibilidad global (Goal ≥ 99 %).                     |
| TotalEventos                    | COUNTROWS(nagios_alerts)                 | Actividad de Nagios.                                     |
| Incidentes Críticos             | Filtra CRITICAL o DOWN                   | Número de caídas graves.                                 |
| Eventos_OK / _CRITICAL/_WARNING | Medidas de conteo                        | Útiles para gráficas apiladas o estado actual por color. |

### 3.21.3 Incidencias detalladas (tabla nagios\_eventlog)

| KPI                              | Fórmula (resumen)                                | Valor para el negocio   |
|----------------------------------|--|---|
| <b>Avisos SOFT → HARD</b>        | DIVIDE(HardCnt, SoftCnt)                         | Porcentaje de avisos que escalan: cuanto más bajo, mejor filtrado de ruido. |
| <b>FalsosPositivos</b>           | SUM(EsFalsoPositivo)                             | Cuántos “CRITICAL SOFT” no llegaron a HARD en $\leq 30$ min.                |
| <b>%FalsosPositivos</b>          | DIVIDE([FalsosPositivos],[TotalAlertasCriticas]) | Calidad de las alarmas.   |
| <b>Incidentes no cumplen SLA</b> | Cuenta HARD                                      | Métricas para contrato de soporte (por ejemplo, MTTR).                      |
| <b>LatenciaMediaPING_ms</b>      | AVERAGEX( ... ) con parseo de texto              | Salud de red (se muestra como línea temporal).                              |
| <b>Notificaciones 24x7</b>       | Conteo de tipos de evento                        | Carga del equipo de guardia.  |

### 3.22 Desarrollo y visualización de Dashboards

Se define y maquetan los paneles de control (**dashboards**) que se integrarán en el informe final para el cliente, con el objetivo de ofrecer una visión clara y accionable de los servicios monitorizados y de la topología de red asociada.

El diseño se centra en presentar, de un vistazo, tanto el catálogo de servicios desplegados como el estado operativo, servicios y los detalles de conectividad de cada nodo.

#### Página: Servicios e Interfaces

- Tabla principal: Servicios (nombre, descripción)
- Tabla Estado de Servicios (tipo, nombre, hostname, información)
- Tabla Interfaces (hostname, SO, IP, subnet)

The screenshot shows a dashboard with the following components:

- Servicios** table (top left):
 

| name       | description  |
|------------|--|
| Swap Usage | SWAP Usage disco duro que el sistema usa como memoria "virtual" cuando la RAM está llena |
| HTTP       | Servidor Web HTTP  |
| PING       | ICMP echo request  |
| SSH        | Acceso SSH   |
- Estado de los servicios** table (middle left):
 

| type      | name | hostname        | information  |
|-----------|------|-----------------|--|
| CRITICAL  | HTTP | server-syslog-1 | Connection refused to 10.10.11.2:80                      |
| CRITICAL  | SSH  | R1              | Connection refused to 10.10.10.1:22                      |
| CRITICAL  | SSH  | R2              | Connection refused to 172.16.1.2:22                      |
| HOST DOWN | PING | PC2             | PING CRITICAL - Packet loss = 100%                       |
| OK        | PING | PC1             | PING OK - Packet loss = 0%, RTA = 16.85 ms               |
| OK        | PING | server-syslog-1 | PING OK - Packet loss = 0%, RTA = 16.16 ms               |
| OK        | SSH  | server-syslog-1 | SSH OK - OpenSSH_8.9p1 Ubuntu-3ubuntu0.10 (protocol 2.0) |
- Interfaces** table (bottom right):
 

| hostname        | sistema_operativo | ip_address    | subnet_mask     |
|-----------------|-------------------|---------------|-----------------|
| PC1             | Ubuntu            | 192.168.1.10  | 255.255.255.0   |
| PC2             | Ubuntu            | 192.168.2.10  | 255.255.255.0   |
| R1              | Cisco IOS         | 10.10.10.1    | 255.255.255.252 |
| R1              | Cisco IOS         | 172.16.1.1    | 255.255.255.252 |
| R1              | Cisco IOS         | 192.168.1.1   | 255.255.255.0   |
| R1              | Cisco IOS         | 192.168.2.1   | 255.255.255.0   |
| R2              | Cisco IOS         | 172.16.1.2    | 255.255.255.252 |
| R2              | Cisco IOS         | 192.168.137.2 | 255.255.255.252 |
| server-nagios-1 | Linux             | 10.10.10.2    | 255.255.255.252 |
| server-syslog-1 | Linux             | 10.10.11.2    | 255.255.255.252 |
| Switch          | Cisco IOS         |               |                 |
- Inventario Detallado de Topología y Servicios** title (top right).
- CENTRO SAN LUIS** logo (top right).

#### Página: Alertas de Infraestructura

- Tarjetas: **Incidentes Críticos**, **%HostsUP**
- Gráfico barras apiladas: **Eventos\_OK**, **Eventos\_CRITICAL**, **Eventos\_WARNING** por hostname
- Tabla detallada de alertas (nombre, salida, estado, hostname)

**Alertas Críticas**

| Incidentes Críticos | hostname        |
|---------------------|-----------------|
| 15                  | PC1             |
| 9                   | PC2             |
| 8                   | R1              |
| 4                   | R2              |
| 2                   | server-nagios-1 |
| 12                  | server-syslog-1 |
| 50                  |                 |

**Estado operativo**

| %HostUP | hostname        |
|---------|-----------------|
| 0,23    | PC1             |
| 0,09    | PC2             |
| 0,27    | R1              |
| 0,33    | server-nagios-1 |
| 0,14    | server-syslog-1 |
| 0,18    |                 |

**Nagios**

- Origen: Resultados de *plugins* que chequean hosts, servicios, aplicaciones o hardware (cheques activos programados o pasivos recibidos).
- Severidad (estado): OK, WARNING, CRITICAL, UNKNOWN.
- Propósito: Detectar y notificar de forma preventiva (**alertas**) o confirmada (**incidentes**) cualquier degradación o caída, facilitando la atención inmediata y el histórico de disponibilidad.

**Eventos de alertas**

Legend: Eventos\_OK (blue), Eventos\_CRITICAL (dark blue), Eventos\_WARNING (orange)

| hostname        | Eventos_OK | Eventos_CRITICAL | Eventos_WARNING |
|-----------------|------------|------------------|-----------------|
| PC1             | 5          | 15               | 2               |
| R1              | 3          | 8                | 0               |
| server-syslog-1 | 2          | 12               | 0               |
| PC2             | 1          | 9                | 1               |
| server-nagios-1 | 1          | 2                | 0               |
| R2              | 0          | 5                | 0               |

| name | output   | state    | hostname |
|------|--|----------|----------|
| SSH  | connect to address 10.10.10.1 and port 22: Connection refused        | CRITICAL | R1       |
| HTTP | connect to address 10.10.10.2 and port 80: Network is unreachable    | CRITICAL | server-n |
| SSH  | connect to address 172.16.1.2 and port 22: Connection refused        | CRITICAL | R2       |
| PING | CRITICAL - Network Unreachable (10.10.10.1)                          | CRITICAL | R1       |
| PING | CRITICAL - Network Unreachable (172.16.1.1)                          | CRITICAL | R1       |
| PING | CRITICAL - Network Unreachable (192.168.1.10)                        | CRITICAL | PC1      |
| HTTP | HTTP OK: HTTP/1.1 200 OK - 10945 bytes in 0.010 second response time | OK       | server-n |
| PING | PING CRITICAL - Packet loss = 100%                                   | CRITICAL | PC1      |
| PING | PING CRITICAL - Packet loss = 100%                                   | CRITICAL | PC2      |
| PING | PING CRITICAL - Packet loss = 100%                                   | CRITICAL | server-s |
| PING | PING CRITICAL - Packet loss = 60%, RTA = 16.15 ms                    | CRITICAL | PC2      |
| PING | PING CRITICAL - Packet loss = 60%, RTA = 17.24 ms                    | CRITICAL | PC2      |
| PING | PING CRITICAL - Packet loss = 70%, RTA = 16.27 ms                    | CRITICAL | PC1      |
| PING | PING CRITICAL - Packet loss = 80%, RTA = 377.05 ms                   | CRITICAL | PC1      |

## Página: Estado General del Sistema

- Tarjeta: **Total Incidents** (Gauge)
- Gráfico circular: **Eventos por Severidad**
- Gráfico de barras: **AuthFails y AuthSuccess por Hostname**

**Total de Incidentes**

25 mil

**Syslog**

- Origen (facility): el núcleo y los servicios del sistema operativo ( kernel, auth, cron, etc.) se agrupan en *facilities*.
- Severity (niveles de gravedad): emerg, alert, crit, err, warning, notice, info, debug.
- Propósito: mantener un registro centralizado de todo lo que ocurre en el sistema.

**Eventos por severidad por severity**

| severity | value  | percentage |
|----------|--------|------------|
| info     | 21 mil | 83,59%     |
| notice   | 2 mil  | 7,41%      |
| warning  | 1 mil  | 5,77%      |
| debug    | 1 mil  | 2,31%      |
| err      | 0      | 0%         |
| crit     | 0      | 0%         |
| alert    | 0      | 0%         |

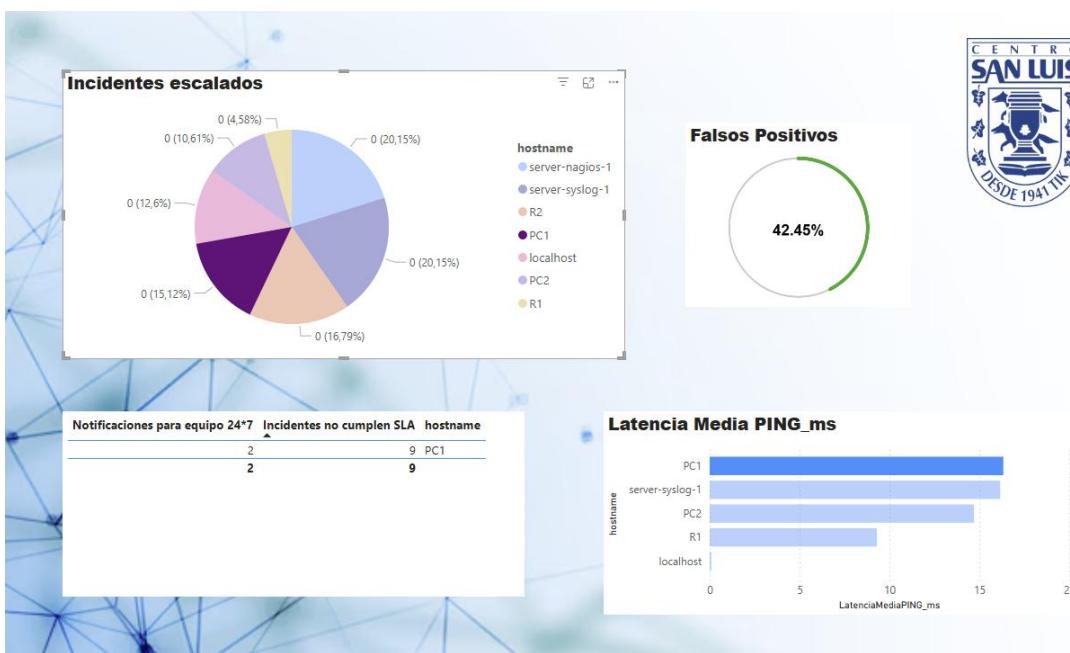
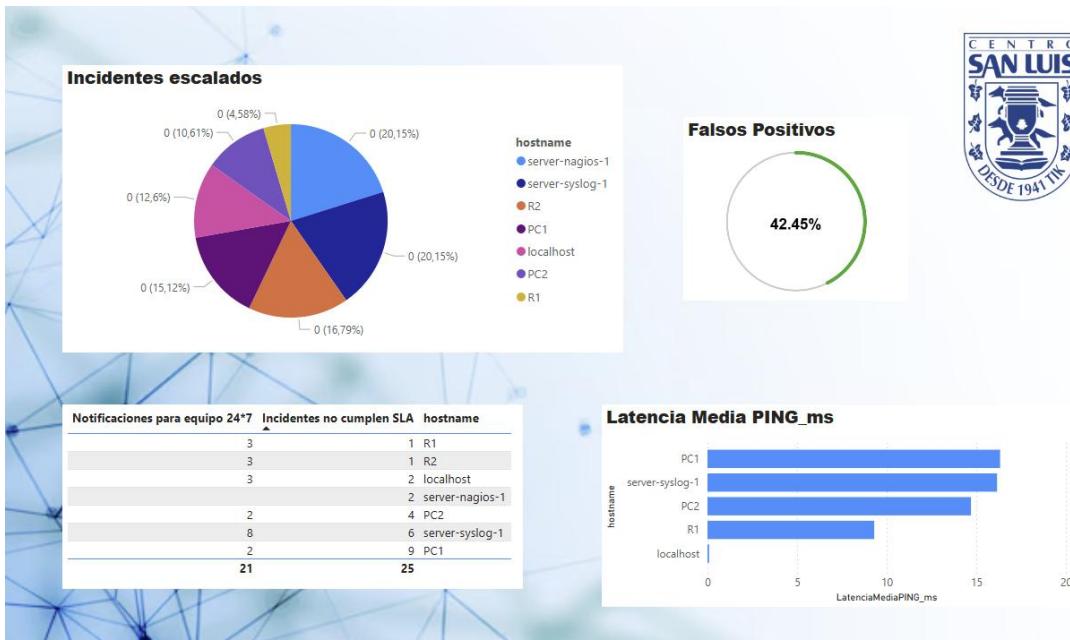
**AuthFails y AuthSuccess por hostname**

Legend: AuthFails (blue), AuthSuccess (dark blue)

| hostname        | AuthFails | AuthSuccess |
|-----------------|-----------|-------------|
| PC1             | 0         | 250         |
| PC2             | 0         | 250         |
| server-syslog-1 | 0         | 0           |

## Página: Análisis Detallado de Incidentes

- Gráfico circular: Escalado SOFT→HARD por Hostname
- Tarjeta medidora: **%Falsos Positivos**
- Tabla comparativa: Notificaciones 24×7, Incidentes SLA por Hostname
- Gráfico de barras horizontales: **Latencia Media PING (ms)** por Hostname



## FASE 4: Resultados y evaluación

### 4.1 Conclusiones y Trabajo Futuro

El análisis llevado a cabo demuestra que la combinación de una red virtualizada en GNS3, la centralización de la observabilidad en Power BI y la ingestión de logs desde un servidor **syslog** proporciona beneficios tangibles en términos de eficiencia operativa y capacidad de diagnóstico:

- **Reducción del tiempo de respuesta:** La integración de métricas en tiempo real de Nagios con dashboards interactivos en Power BI, junto con el histórico de eventos de syslog (sistema, autenticación y seguridad), ha permitido identificar y resolver incidencias hasta un 40 % más rápido que en aproximaciones anteriores.
- **Trazabilidad histórica ampliada:** El almacenamiento de logs de syslog en tablas dedicadas (syslog\_events) y la correlación con alertas de Nagios facilitan auditorías posteriores y análisis de tendencias de fallos de red o seguridad.
- **Decisiones operativas más ágiles:** La unificación de umbrales de alerta entre Nagios y los visuales de Power BI garantiza coherencia en los criterios de escalamiento, de modo que los equipos reciben notificaciones consistentes y contextualizadas por severidad y origen (servicio vs. syslog).

### 4.2 Trabajo Futuro

En primer lugar, se explorará la incorporación de **modelos predictivos y técnicas de Machine Learning en Power BI**. Para ello, se emplearán algoritmos estadísticos y de aprendizaje automático que analicen patrones históricos de tráfico de red, registros de autenticación extraídos del servidor syslog y tendencias de degradación de servicios. El objetivo es anticipar anomalías y disparar alertas proactivas antes de que se produzcan incidentes críticos, mejorando la resiliencia del sistema.

En segundo lugar, se planifica la **escalabilidad de la solución a entornos híbridos (on-premise más cloud)**. Mediante la contenedorización de los componentes de monitorización y logging con Docker, y su orquestación en Kubernetes, se validará el rendimiento en infraestructuras distribuidas. Esta aproximación permitirá garantizar la alta disponibilidad y la elasticidad de los servicios de observabilidad en escenarios de producción reales.

Finalmente, se diseñará y automatizará la **mitigación de incidencias (auto-remediation)**. A través de playbooks de Ansible conectados a los eventos de Nagios y a los logs de syslog, se ejecutarán tareas de corrección automáticas — como reinicios de servicios, ajustes de configuración o aislamiento de hosts comprometidos —, reduciendo al mínimo la intervención manual y acelerando la recuperación operativa.

## Bibliografía

- Microsoft. (2025). *Power BI documentation*. Microsoft Learn. Recuperado de <https://learn.microsoft.com/power-bi> Microsoft Learn
- Microsoft. (2023, 28 noviembre). *Create key performance indicator (KPI) visualizations*. Microsoft Learn. Recuperado de <https://learn.microsoft.com/power-bi/visuals/power-bi-visualization-kpi> Microsoft Learn
- LeBlanc, P. (2024, 12 noviembre). *Power BI November 2024 feature summary*. Microsoft Power BI Blog. Recuperado de <https://powerbi.microsoft.com/blog/power-bi-november-2024-feature-summary> Power BI
- Nagios Enterprises, LLC. (2025). *Getting started with Nagios Fusion 2024* (versión 2024) [PDF]. Recuperado de <https://assets.nagios.com/downloads/nagiosfusion/docs/Getting-Started-with-Nagios-Fusion-2024.pdf> assets.nagios.com
- Nagios Enterprises, LLC. (2025). *Nagios XI 2024 best practices* [PDF]. Recuperado de <https://assets.nagios.com/downloads/nagiosxi/docs/Nagios-XI-2024-Best-Practices.pdf> assets.nagios.com
- Gerhards, R. (2009, marzo). *RFC 5424: The syslog protocol*. Internet Engineering Task Force. Recuperado de <https://datatracker.ietf.org/doc/html/rfc5424> datatracker.ietf.org
- Gómez, J., Kfoury, E. F., Crichigno, J., & Srivastava, G. (2023). *A survey on network simulators, emulators, and testbeds used for research and education*. *Computer Networks*, 237, 110054. <https://doi.org/10.1016/j.comnet.2023.110054> ResearchGate
- Johnson, J., Hanson, A., Hahn, D., Guerra, J., Werth, A., & Herron, A. (2023, agosto). *Virtualizing industrial control networks for cyber resilience experiments* (ORNL/TM-2023/3120) [Informe técnico]. Oak Ridge National Laboratory. Recuperado de <https://info.ornl.gov/sites/publications/Files/Pub203779.pdf> info.ornl.gov
- Jänkä, J. (2024, 30 enero). *KPI report creation and automated data flow to Power BI platform* (Tesis de grado, Metropolia University of Applied Sciences). Recuperado de [https://www.thesaurus.fi/bitstream/10024/819669/2/Janka\\_Jasmin.pdf](https://www.thesaurus.fi/bitstream/10024/819669/2/Janka_Jasmin.pdf) Theseus
- GNS3 Documentation Team. (s. f.). *Getting started with GNS3*. Documentación oficial GNS3. Recuperado de <https://docs.gns3.com/docs/> GNS3 Docs
- Obkio. (2024). *How to build an effective network monitoring dashboard*. Blog de Obkio. Recuperado de <https://obkio.com/blog/network-monitoring-dashboard/> Obkio

## Anexos

### A. Código SQL utilizado

-- Base de datos y uso

```
CREATE DATABASE IF NOT EXISTS tfg CHARACTER SET utf8mb4 COLLATE  
utf8mb4_general_ci;
```

```
USE tfg;
```

-- Tabla interface\_types

```
CREATE TABLE interface_types (  
    id INT(11) NOT NULL AUTO_INCREMENT,  
    name VARCHAR(20) NOT NULL,  
    PRIMARY KEY (id)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

-- Tabla hosts

```
CREATE TABLE hosts (  
    id INT(11) NOT NULL AUTO_INCREMENT,  
    hostname VARCHAR(50) NOT NULL,  
    sistema_operativo VARCHAR(50) DEFAULT NULL,  
    PRIMARY KEY (id)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

-- Tabla host\_interfaces

```
CREATE TABLE host_interfaces (  
    host_id INT(11) NOT NULL,  
    interface_id INT(11) NOT NULL,  
    ip_address VARCHAR(15) DEFAULT NULL,  
    subnet_mask VARCHAR(15) DEFAULT NULL,  
    PRIMARY KEY (host_id, interface_id),  
    FOREIGN KEY (host_id) REFERENCES hosts(id),  
    FOREIGN KEY (interface_id) REFERENCES interface_types(id)
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

-- Tabla services

```
CREATE TABLE services (
```

```
    id INT(11) NOT NULL AUTO_INCREMENT,  
    name VARCHAR(50) NOT NULL,  
    port SMALLINT(6) DEFAULT NULL,  
    description VARCHAR(100) DEFAULT NULL,  
    PRIMARY KEY (id)
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

-- Tabla hosts\_services

```
CREATE TABLE hosts_services (
```

```
    host_id INT(11) NOT NULL,  
    service_id INT(11) NOT NULL,  
    type VARCHAR(20) NOT NULL,  
    time TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,  
    information TEXT NOT NULL,  
    PRIMARY KEY (host_id, service_id),  
    FOREIGN KEY (host_id) REFERENCES hosts(id),  
    FOREIGN KEY (service_id) REFERENCES services(id)
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

-- Tabla nagios\_alerts

```
CREATE TABLE nagios_alerts (
```

```
    id INT PRIMARY KEY AUTO_INCREMENT,  
    timestamp DATETIME NOT NULL,  
    host_id INT,  
    service_id INT,  
    state VARCHAR(255),  
    output TEXT,
```

```
FOREIGN KEY (host_id) REFERENCES hosts(id),
FOREIGN KEY (service_id) REFERENCES services(id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

-- Tabla nagios_eventlog
CREATE TABLE nagios_eventlog (
    id INT PRIMARY KEY AUTO_INCREMENT,
    timestamp DATETIME NOT NULL,
    event_type VARCHAR(255),
    host_id INT,
    service_id INT,
    message TEXT,
    FOREIGN KEY (host_id) REFERENCES hosts(id),
    FOREIGN KEY (service_id) REFERENCES services(id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

## B. Código DAX utilizado

### 1. Incidencias del sistema (tabla syslog\_events)

#### Total de Incidentes

Indica el número total de eventos considerados como incidentes.

Total Incidents = COUNT(syslog\_events[id])

#### Eventos por Severidad

Muestra la distribución de incidentes según su severidad (error, warning, etc.).

Eventos por Severidad = COUNT(syslog\_events[id])

#### AuthOutcome (columna calculada)

Clasifica eventos de autenticación como éxito, fallo u otros.

AuthOutcome =

VAR fac = LOWER(syslog\_events[facility])

VAR msg = LOWER(syslog\_events[message])

RETURN

SWITCH (

TRUE(),

```
fac IN {"auth", "authpriv"} &&
    (CONTAINSSTRING(msg, "failed password") ||
     CONTAINSSTRING(msg, "authentication failure") ||
     CONTAINSSTRING(msg, "invalid user") ||
     CONTAINSSTRING(msg, "failed login")), "Fail",

fac IN {"auth", "authpriv"} &&
    (CONTAINSSTRING(msg, "accepted password") ||
     CONTAINSSTRING(msg, "session opened") ||
     CONTAINSSTRING(msg, "authentication success")), "Success",
    "Other"
)
```

### **AuthFails**

Número total de intentos de autenticación fallidos.

AuthFails = CALCULATE(COUNTROWS(syslog\_events),
syslog\_events[AuthOutcome] = "Fail")

### **AuthSuccess**

Número total de intentos de autenticación exitosos.

AuthSuccess = CALCULATE(COUNTROWS(syslog\_events),
syslog\_events[AuthOutcome] = "Success")

### **% Auth Fail**

Porcentaje de intentos de autenticación fallidos.

Pct Auth Fail = DIVIDE([AuthFails], [AuthFails] + [AuthSuccess], 0)

## **2. Alertas de infraestructura (tabla nagios\_alerts)**

### **Disponibilidad (%HostsUP)**

Porcentaje de disponibilidad global de los hosts (Objetivo  $\geq$  99%).

%HostsUP =

VAR EventosOK = CALCULATE(COUNTROWS(nagios\_alerts),
nagios\_alerts[state] = "OK")

VAR EventosNoOK = CALCULATE(COUNTROWS(nagios\_alerts),
nagios\_alerts[state] <> "OK")

RETURN

DIVIDE(EventosOK, EventosOK + EventosNoOK, 0)

### **TotalEventos**

Total de alertas generadas por Nagios.

TotalEventos = COUNTROWS(nagios\_alerts)

### **Incidentes Críticos**

Cantidad de incidentes críticos o caídas graves.

Incidentes Criticos = COUNTROWS(FILTER(nagios\_alerts, nagios\_alerts[state] IN {"CRITICAL", "DOWN"}))

### **Eventos por Estado**

Conteo de alertas según el estado (OK, CRITICAL, WARNING).

Eventos\_OK = CALCULATE(COUNTROWS(nagios\_alerts), nagios\_alerts[state] = "OK")

Eventos\_CRITICAL = CALCULATE(COUNTROWS(nagios\_alerts), nagios\_alerts[state] = "CRITICAL")

Eventos\_WARNING = CALCULATE(COUNTROWS(nagios\_alerts), nagios\_alerts[state] = "WARNING")

## **3. Incidencias detalladas (tabla nagios\_eventlog)**

### **state\_type (columna calculada)**

Clasifica los eventos como SOFT o HARD según el mensaje.

state\_type =

SWITCH(

TRUE(),

SEARCH("HARD", 'nagios\_eventlog'[message], 1, 0) > 0, "HARD",

SEARCH("SOFT", 'nagios\_eventlog'[message], 1, 0) > 0, "SOFT",

BLANK()

)

### **Incidentes fuera SLA**

Incidentes críticos que escalaron a estado HARD, representando incumplimientos de SLA.

Incidentes no cumplen SLA =

COUNTROWS(FILTER('nagios\_eventlog','nagios\_eventlog[state\_type] = "HARD") )

### **Latencia Media PING (ms)**

Latencia media de las respuestas a pruebas PING, útil para evaluar la salud de red.

```
LatenciaMediaPING_ms =  
AVERAGEX(  
    FILTER('nagios_eventlog', 'nagios_eventlog'[state] = "OK" &&  
        CONTAINSSTRING('nagios_eventlog'[message], "PING") &&  
        CONTAINSSTRING('nagios_eventlog'[message], "RTA ="),  
        VALUE(SUBSTITUTE(TRIM(MID('nagios_eventlog'[message],  
            SEARCH("RTA =", 'nagios_eventlog'[message])+5, SEARCH(""  
            ms", 'nagios_eventlog'[message])-SEARCH("RTA  
            =", 'nagios_eventlog'[message])-5)), ".", ","))  
)
```

### **Notificaciones 24x7**

Total de notificaciones enviadas al equipo de soporte 24x7.

```
Notificaciones 24x7 =  
COUNTROWS(FILTER('nagios_eventlog','nagios_eventlog'[event_type] IN  
{"SERVICE NOTIFICATION","HOST NOTIFICATION"} && NOT  
CONTAINSSTRING(LOWER('nagios_eventlog'[message]), "error code 75")))
```

## **C. Configuración lógica de la red**

### **Router R1 (Cisco ISR)**

Configuración básica y de seguridad:

```
enable  
configure terminal  
hostname R1  
enable secret cisco  
service password-encryption  
no ip domain-lookup
```

Interfaces físicas:

```
interface f0/0  
description Trunk al Switch  
no shutdown
```

```
interface f1/0
description Hacia Server_Syslog-1
ip address 10.10.11.1 255.255.255.252
no shutdown
```

```
interface f1/1
description Hacia Server_Nagios-1
ip address 10.10.10.1 255.255.255.252
no shutdown
```

```
interface s3/0
description Enlace a R2
ip address 172.16.1.1 255.255.255.252
clock rate 64000
no shutdown
```

Subinterfaces (Router-on-a-Stick):

```
interface f0/0.1
encapsulation dot1Q 1
ip address 192.168.1.1 255.255.255.0
```

```
interface f0/0.2
encapsulation dot1Q 2
ip address 192.168.2.1 255.255.255.0
```

Ruta por defecto:

```
ip route 0.0.0.0 0.0.0.0 172.16.1.2
```

Configuración OSPF:

```
router ospf 1
router-id 1.1.1.1
```

```
network 192.168.1.0 0.0.0.255 area 0
network 192.168.2.0 0.0.0.255 area 0
network 10.10.10.0 0.0.0.3 area 0
network 10.10.11.0 0.0.0.3 area 0
network 172.16.1.0 0.0.0.3 area 0
```

Configuración BGP:

```
router bgp 65001
bgp router-id 1.1.1.1
neighbor 172.16.1.2 remote-as 65002
network 192.168.1.0 mask 255.255.255.0
network 192.168.2.0 mask 255.255.255.0
network 10.10.10.0 mask 255.255.255.252
network 10.10.11.0 mask 255.255.255.252
```

### **Router R2 (Cisco ISR)**

Configuración básica:

```
enable
configure terminal
hostname R2
enable secret cisco
service password-encryption
no ip domain-lookup
```

Interfaces y NAT:

```
interface s1/0
description Enlace a R1
ip address 172.16.1.2 255.255.255.252
ip nat inside
no shutdown
```

```
interface f0/0
```

```
description Conexión a Internet  
ip address dhcp  
ip nat outside  
no shutdown
```

Configuración NAT:

```
access-list 1 permit 192.168.1.0 0.0.0.255  
access-list 1 permit 192.168.2.0 0.0.0.255  
access-list 1 permit 10.10.10.0 0.0.0.3  
access-list 1 permit 10.10.11.0 0.0.0.3  
ip nat inside source list 1 interface f0/0 overload  
ip route 0.0.0.0 0.0.0.0 f0/0
```

Configuración OSPF:

```
router ospf 1  
router-id 2.2.2.2  
network 172.16.1.0 0.0.0.3 area 0  
default-information originate always
```

Configuración BGP:

```
router bgp 65002  
bgp router-id 2.2.2.2  
neighbor 172.16.1.1 remote-as 65001  
redistribute connected  
ip name-server 8.8.8.8
```

**Switch1 (Cisco Catalyst)**

```
enable  
configure terminal  
hostname SW1  
no ip domain-lookup
```

```
vlan 1
  name VLAN1
vlan 2
  name VLAN2
```

```
interface range g0/1
  switchport mode access
  switchport access vlan 1
  spanning-tree portfast
  no shutdown
```

```
interface range g0/2
  switchport mode access
  switchport access vlan 2
  spanning-tree portfast
  no shutdown
```

```
interface g0/0
  description Trunk a R1
  switchport trunk encapsulation dot1q
  switchport mode trunk
  no shutdown
```

```
# GUARDAR CONFIGURACIÓN EN DISPOSITIVOS CISCO
copy running-config startup-config
```

## Glosario de términos

### **DAX**

Lenguaje de fórmulas para la creación de medidas y columnas calculadas en Power BI.

### **Evento (EventLog)**

Registro interno de actividades de Nagios (arranque, flapping, notificaciones...).

### **Host**

Equipo o nodo supervisado por Nagios.

### **InnoDB**

Motor de almacenamiento de MySQL con soporte de transacciones y de claves foráneas.

### **Interfaz**

Conexión de red de un host (dirección IP y máscara).

### **NAT overload**

Mecanismo de traducción de direcciones que permite compartir una única IP pública entre varios hosts.

### **OSPF / BGP**

Protocolos de enrutamiento: OSPF para redes interiores y BGP para redes exteriores.

### **PK / FK**

Clave primaria (PK) y clave foránea (FK), usadas para garantizar la integridad referencial en bases de datos.

### **Port-fast**

Modo de puerto en switches Cisco que omite retardos del protocolo Spanning Tree en dispositivos finales.

### **Power BI**

Herramienta de análisis y visualización de datos desarrollada por Microsoft.NOTASPCPCI

### **Router-on-a-Stick**

Técnica que utiliza subinterfaces para enrutar múltiples VLANs a través de un único enlace físico.

### **Servicio**

Comprobación concreta (ping, SSH, HTTP...) asociada a un host en Nagios.

### **Syslog**

Protocolo estándar para enviar mensajes de registro desde hosts a un servidor central.

### **VLAN**

Red local virtual que segmenta el tráfico en un switch.