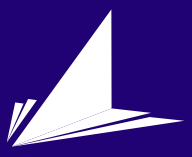

Escuela
Politécnica
Superior

Sistema vía Web de monitorización de red con detección de anomalías

Máster Universitario en Ingeniería en Telecomunicación

Trabajo Fin de Máster

Autor:
Francisco José Olmo Valverde
Tutor:
Jaume Aragonés Ferrero

Universitat d'Alacant
Universidad de Alicante

Agosto 2025

Sistema vía Web de monitorización de red con detección de anomalías

Implementación de un sistema web de supervisión y alerta de red, probado en entornos simulados con GNS3

Autor
Francisco José Olmo Valverde

Tutor
Jaume Aragonés Ferrero
Lenguajes y Sistemas Informáticos

Máster Universitario en Ingeniería en Telecomunicación

Escuela
Politécnica
Superior

Universitat d'Alacant
Universidad de Alicante

ALICANTE, Agosto 2025

Resumen

En la actualidad, la infraestruturas de red presentan un grado creciente de complejidad debido a la diversidad de dispositivos y servicios que las componen. Esta situación exige herramientas avanzadas que permitan supervisar el estado de la red en tiempo real, detectar incidencias de manera temprana y facilitar la toma de decisiones por parte de los administradores.

El siguiente Trabajo Final de Máster se aborda el diseño e implementación de un sistema web de monitorización de red con capacidad de detección de anomalías. El objetivo principal es una plataforma integral que permita obtener la información relevante de los dispositivos de red, analizar su comportamiento y representar los resultados en una interfaz visual clara e interactiva.

La solución se ha desarrollado siguiendo una arquitectura cliente-servidor. El backend, implementado en Python mediante el framework FastAPI, establece conexión con router Cisco del emulador GNS3 utilizando el protocolo SSH (a través de la librería Paramiko) para extraer métricas como el estado de las interfaces, tráfico de entrada y salida, utilización de CPU y memoria, así como cambios de configuración en los dispositivos. Por su parte, el frontend se ha construido con Vue.js Bootstrap y Chart.js, ofreciendo visualización dinámica de los datos mediante gráficos en tiempo real, tablas interactivas y barras de progreso que facilitan la interpretación de los parámetros monitorizados.

Además, el sistema incorpora mecanismo de seguridad y gestión de usuarios mediante autenticación basada en JSON Web Tokens (JWT) Lo que garantiza el acceso restringido a la aplicación. También se ha implementado un módulo de generación automática de informes en formato PDF y su envío por correo electrónico en caso de detección de errores.

Los resultados obtenidos evidencia que la solución propuesta cumple los objetivos planteados: permite monitorizar en tiempo real distintos dispositivos de red, detectar anomalías en su funcionamiento y presentar la información de forma accesible y comprensible. Este sistema constituye una herramienta de apoyo útil para la gestión de redes de telecomunicaciones, con potencial de apliación hacia entornos más complejos.

Abstract

In today's context, network infrastructures present an increasing degree of complexity due to the diversity of devices and services they encompass. This situation requires advanced tools capable of continuously supervising the network status in real time, detecting incidents at an early stage, and facilitating decision-making for administrators.

This Master's Thesis addresses the design and implementation of a web-based network monitoring system with anomaly detection capabilities. The main objective is to provide an integral platform that allows the retrieval of relevant information from network devices, the analysis of their behavior, and the representation of results through a clear and interactive visual interface.

The solution has been developed following a client-server architecture. The backend, implemented in Python using the FastAPI framework, connects to Cisco routers in the GNS3 emulator via the SSH protocol (through the Paramiko library) to extract metrics such as interface status, inbound and outbound traffic, CPU and memory utilization, as well as configuration changes in the devices. The frontend, developed with Vue.js, Bootstrap, and Chart.js, offers a dynamic visualization of the data through real-time charts, interactive tables, and progress bars that facilitate the interpretation of monitored parameters.

In addition, the system integrates security mechanisms and user management through JSON Web Token (JWT) authentication, ensuring restricted access to the application. An automatic report generation module in PDF format and its delivery via email in case of error detection has also been implemented.

The results obtained demonstrate that the proposed solution meets the established objectives: it enables real-time monitoring of different network devices, detects anomalies in their operation, and presents the information in an accessible and understandable manner. This system constitutes a valuable support tool for the management of telecommunication networks, with potential for extension to more complex environments.

Agradecimientos

Índice general

Índice de figuras

Índice de tablas

Índice de Códigos

1. Introducción

1.1. Motivación

La gestión de redes de telecomunicaciones se enfrenta a un escenario cada vez más complejo, en el que la administración manual de dispositivos resulta ineficiente y propensa a errores. La creciente demanda de disponibilidad y rapidez en la resolución de incidencias ha impulsado la necesidad de automatizar procesos de monitorización y configuración. En este contexto, el uso de APIs para interactuar con dispositivos de red se ha convertido en un elemento clave para agilizar tareas de supervisión, mejorar la interoperabilidad y reducir los tiempos de respuesta de los administradores.

La motivación de este proyecto surge precisamente de esta necesidad: desarrollar un sistema que integre tecnologías web y APIs para obtener información en tiempo real de los routers y automatizar tanto la supervisión como ciertas operaciones de gestión. Tal y como indican trabajos recientes en la literatura, la incorporación de automatización y monitorización mejora la capacidad de respuesta y refuerza la fiabilidad de las infraestructuras (Schummer, 2024).

1.2. Objetivo General

El objetivo general de este Trabajo final de Máster es el desarrollo de un sistema web de monitorización de red con detección de anomalías, capaz de supervisar de manera centralizada y remota distintos dispositivos Cisco en un entorno de laboratorio simulado mediante GNS3. La plataforma debe ser capaz de:

- Extraer métricas relevantes de los routers (estado de interfaces, tráfico, errores, uso de CPU y memoria, cambios de configuración).
- Presentar los resultados mediante una interfaz gráfica clara e interactiva en tiempo real.
- Incorporar funcionalidades de seguridad, autenticación de usuarios y generación deinformes automáticos en PDF con envío por correo electrónico en caso de incidencias.

Este enfoque está en línea con distintos trabajos realizados quienes demuestran aplicabilidad de automatización para mejorar la detección de anomalías y reforzar la monitorización de sistemas de comunicaciones (Schummer, 2024).

1.3. Contenidos Previos

El proyecto se apoya en un conjunto de contenidos previos adquiridos durante el máster en telecomunicaciones y el grado en ingeniería en sonido e imagen en telecomunicaciones y en el uso de herramientas de amplio reconocimeinto en el ámbito de la informática y las telecomunicaciones:

- Emulador GNS3, ampliamente utilizado para simular redes basadas en routers Cisco, proporcionando un entorno controlado para pruebas y validación (Technologies, 2024).
- Protocolo SSH, estándar seguro para la administración remota de dispositivos de red (RFC 4251) (Ylonen y Lonvick, 2006).
- FastAPI, framework de Python para el desarrollo de APIs rápidas y eficientes, utilizado en el backend (Ramírez, 2024).
- Vue.js y Bootstrap, frameworks modernos de desarrollo web que permiten construir interface dinámicas e intuitivas en el frontend (Community, 2024), (Team, 2024).
- Chart.js, librería especializada en la visualización de datos en tiempo real mediante gráficos (Developers, 2024).
- JSON Web Tokens(JWT). tecnología de autenticación que asegura la gestión de usuarios y el acceso restringido en aplicaciones distribuidas (Auth0, 2024).

Gracias a esta combinación de conocimientos y herramientas, se ha podido desarrollar un sistema que constituye una herramienta de apoyo para la gestión de redes de telecomunicaciones, con potencial de evolución a escenarios de mayor complejidad.

2. Estudio de Viabilidad

2.1. Análisis DAFO

El análisis DAFO realizado en este trajo se centra en evaluar los factores internos y externos que influyen en el desarrollo del sistema web.

Dentro de las **fortalezas**, se destaca el uso de stack tecnológico moderno y eficiente (FastAPI en el backend, Vue y Bootstrap en el frotend y Chart.js para la representación gráfica), lo que permite un desarrollo modular rápido y con gran capacidad de visualización en tiempo real. Asimismo, la integración mediante conexiones SSH a routers Cisco otorga al sistema una aplicación práctica inmediata.

En cuanto a las **debilidades**, se identifican la experiencia limitada en el uso de algoritmos avanzados de detección de anomalías, la dependencia inicial de un único fabricante (Cisco) y la necesidad de mayor tiempo y recursos para ampliar la solución hacia entornos más complejos.

Respecto a las **oportunidades**, el sistema se orienta principalmente principalmente a un entorno académico, donde puede servir como apoyo al aprendizaje de conceptos avanzados de redes y monitorización. Esta herramienta facilita la compresión práctica de la interpretación de métricas de tráfico y la detección de anomalías, aspectos que resultan especialmente útiles en ingeniería de telecomunicación. Asimismo, puede constituir a una base sobre la cual los estudiantes experimenten con nuevas técnicas de análisis.

Por último, entre las **amenazas** se encuentran la rápida evolución de las ciberamenazas y las soluciones comerciales ya consolidadas en el mercado (Nagios, Zabbix, Prometheus, entre otras),que pueden limitar la implantación del sistema en entornos productivos. Además, los posibles cambios en las interfaces de configuración de los fabricantes podrían afectar a la compatibilidad a largo plazo.

La Figura 2.1 muestra de forma esquemática este análisis aplicado al proyecto desarrollado.

Análisis DAFO

Sistema web de monitorización de red con detección de anomalías · TFM — Francisco José Olmo Valverde

FORTALEZAS Interno · Positivo

- Stack moderno: FastAPI, Vue, Bootstrap, Chart.js.
- Autenticación JWT e integración modular.
- Datos casi en tiempo real con routers Cisco (SSH).
- Alto valor académico y aplicabilidad inmediata.

DEBILIDADES Interno · Negativo

- Experiencia limitada en detección de anomalías (ML).
- Recursos de laboratorio y tiempo ajustados.
- Curva de aprendizaje en SSH/IOS de Cisco.
- Portabilidad inicial a un solo fabricante.

OPORTUNIDADES Externo · Positivo

- Orientación académica: apoyo al aprendizaje en ingeniería de telecomunicación.
- Facilita la comprensión práctica de métricas de tráfico y detección de anomalías.
- Herramienta base para experimentar con nuevas técnicas de análisis en el aula.
- Potencial evolución hacia entornos docentes más complejos.

AMENAZAS Externo · Negativo

- Rápida evolución de las ciberamenazas.
- Soluciones comerciales consolidadas (Nagios, Zabbix, Prometheus, etc.).
- Limitaciones para implantación en entornos productivos frente a grandes competidores.
- Cambios en las interfaces de configuración que afecten a la compatibilidad.

Figura 2.1: Análisis DAFO del proyecto

3. Objetivos

En este capítulo se definen los objetivos que guían el desarrollo del presente trabajo. Estos objetivos se han clasificado en principal, secundarios y transversales, con el fin delimitar claramente el alcance del proyecto.

3.1. Objetivo principal

El objetivo principal del trabajo es desarrollar un sistema web integral de monitorización de red con capacidad de detectar anomalías en su funcionamiento. Este sistema permitirá obtener información en tiempo real sobre el estado de los routers y sus interfaces, procesarla de forma eficiente y presentarla en una interfaz gráfica amigable para el usuario final.

3.2. Objetivos secundarios

De ese objetivo principal se derivan los siguientes objetivos secundarios:

- Diseñar e implementar un backend basado en FastAPI que permita la comunicación con los dispositivos de red mediantes conexiones SSH.
- Desarrollar un frontend web con Vue.js, Bootstrap y Chart.js para la visualización de datos en tiempo real, incluyendo el estado de interfaces, tráfico y errores detectados.
- Implementar un mecanismo de detección de anomalías (por ejemplo, mediante correo electrónico con informes en PDF) cuando se detecten incidencias relevantes en la red.
- Garantizar la seguridad y control de acceso al sistema mediante autenticación de usuarios.

3.3. Objetivos transversales

Además, el proyecto contribuye al desarollo de una serie de competencias transversales:

- Capacidad de investigación y aprendizaje autónomo. Adquisición de conocimientos avanzados sobre protocolos de red, monitorización y tecnologías web.
- Integración de diferentes areas de conocimiento. Combinación de ingeniería de telecomunicación, programación web, seguridad informática y administración de redes.
- Aplicación práctica de herramientas profesionales. Uso de tecnologías modernas (FastAPI, Vue.js, Bootstrap, Chart.js) y entornos de virtualización/laboratorio para pruebas.

- Trabajo orientado a la innovación. Desarrollo de un sistema con valor añadido mediante la detección de anomalías, más allá de la mera monitorización estática.

4. Estado del Arte

En este capítulo se presentan las principales soluciones existentes en el ámbito de la monitorización de redes. El objetivo es identificar sus características más relevantes, así como sus ventajas y limitaciones, de manera que sirvan como referencia para el diseño del sistema propuesto en este trabajo.

4.1. Nagios: Sistema de Monitorización de Redes

Nagios es una de las soluciones más utilizadas en el ámbito de la monitorización de infraestructuras IT. Su capacidad de supervisar redes, servidores y aplicaciones la convierten en una opción versátil y ampliamente adoptada en entornos empresariales.

Uno de los principales productos de Nagios es Nagios XI, el cual proporciona un conjunto de herramientas avanzadas para la monitorización de infraestructuras críticas (Nagios Enterprises, 2014). Este software perminte supervisar el estado de aplicaciones, servicios, sistemas operativos y protocolos de red. Además, cuenta con un sistema de alertas en tiempo real que notifica a los administradores mediante correo electrónico, SMS o mensajería instantánea. Tiene la capacidad de generar informes detallados sobre el historial de fallos, análisis de tendencias y planificación de capacidad es otra de sus características más destacadas. Una de las fortalezas es su interfaz web, la cual permite a los usuarios personalizar paneles de control para visualizar métricas clave (Nagios Enterprises, 2014).

Figura 4.1: Ejemplo visual de Nagios XI (Nagios Enterprises, 2014)

Otro componente relevante es Nagio Fusion, diseñado para proporcionar una vista consolidad de múltiples instancias de Nagios (Nagios Enterprises, 2014). A gran escala, esta herramienta resulta especialmente útil, ya que permite gestionar diferentes servidores de monitoreo desde una única plataforma (Nagios Enterprises, 2014).

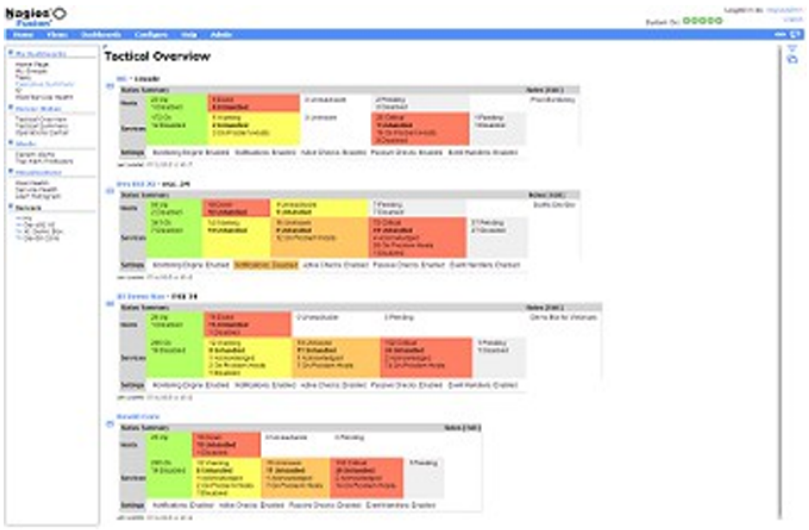

Figura 4.2: Ejemplo visual de Nagios Fusion (Nagios Enterprises, 2014).

En el ámbito de la gestión de incidentes, Nagios Incident Manager facilita la administración de alertas y eventos críticos detectados por Nagios XI (Nagios Enterprises, 2014). Su integración permite la creación y la asignación de tickets en tiempo real, fomentando la colaboración entre equipos IT y agilizando la resolución de problemas.

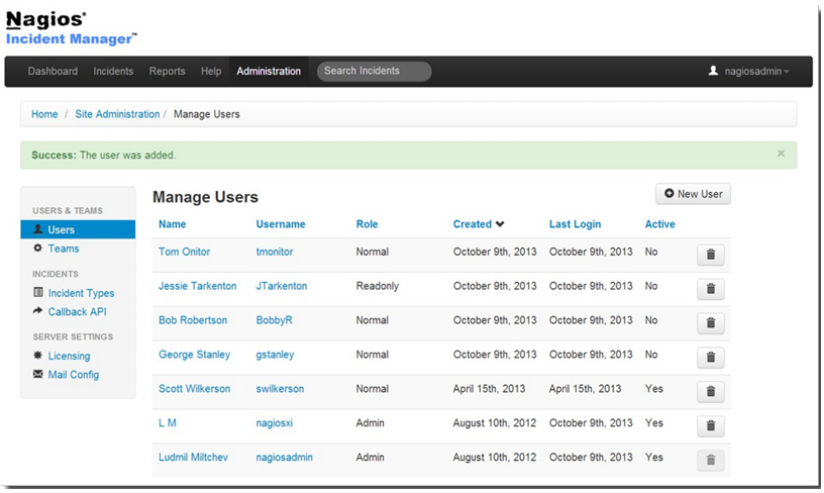

Figura 4.3: Ejemplo visual de Nagios Incident Manager (Nagios Enterprises, 2014).

Por otro lado, Nagios Network Analyzer se enfoca en el análisis de tráfico de red (Na-

gios Enterprises, 2014). Esta herramienta permite supervisar el uso del ancho de banda y detectar patrones de tráfico que pueden indicar problemas o posibles amenazas de seguridad. Su compatibilidad con tecnologías como NetFlow y sFlow permite recopilar información detallada sobre el tráfico en la red, alertando a los administradores en caso de picos inusuales o comportamientos anómalos. Además, su capacidad de generar gráficos avanzados facilita la interpretación de datos y el diagnóstico de problemas en la red (Nagios Enterprises, 2014).

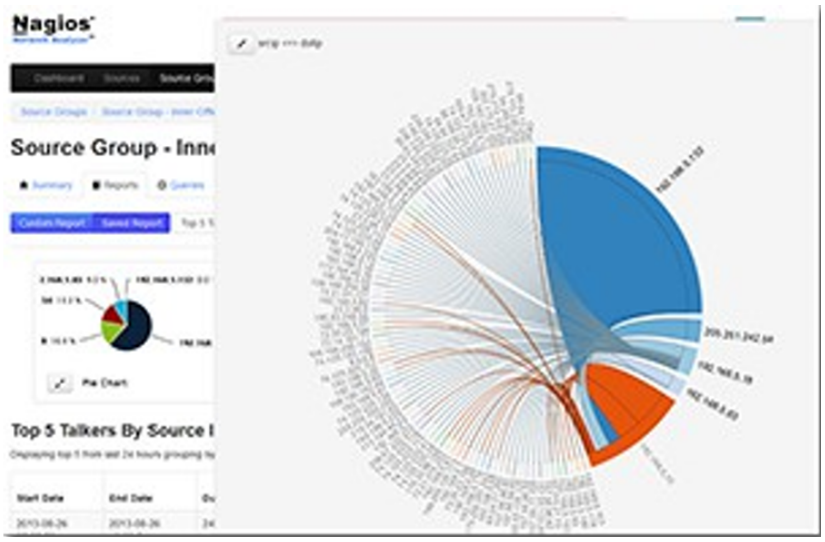

Figura 4.4: Ejemplo Visual de Nagios Network Analyzer (Nagios Enterprises, 2014).

Dentro del contexto del Trabajo final de Máster, Nagios representa un modelo robusto que puede servir de referencia. En particular, Nagios XI y Nagios Network Analyzer, la posibilidad de personalizar dashboards, gestionar alertas en tiempo real y analizar tráfico de red son clave en la planificación y el diseño del trabajo.

Sin embargo, también es importante indicar las limitaciones del software. Aunque su sistema de monitorización es configurable, puede requerir una curva de aprendizaje elevada, especialmente para administradores sin experiencia previa. Además, su modelo basado en plugins y extensiones puede hacer que la integración de otros sitemas requiera una configuración adicional.

4.2. Zabbix: Sistema de Monitorización de Redes

Zabbix es una plataforma de monitorización de código abierto que ha sido ampliamente adoptada en la industria debido a su flexibilidad, escalabilidad y capacidad de integración con diversas tecnologías.

4.3. Prometheus: Sistema de Monitorización de Redes

Prometheus es una plataforma de monitorización y alerta de código abierto diseñada para la recolección y análisis de métricas en entornos dinámicos.

4.4. SolarWinds Network Performance Monitor

SolarWinds Network Performance Monitor (NPM) es una solución comercial diseñada para proporcionar una visión integral del estado y rendimiento de infraestructuras de redes de comunicaciones.

4.5. PRTG Network Monitor

PRTG Network Monitor, desarrollado por Paessler AG, es una plataforma de monitorización de redes que destaca por su enfoque integral, intuitivo y escalable.

4.6. Comparativa de herramientas

Con el fin de obtener una visión global, en la Figura 4.5 se presenta una tabla comparativa que sintetiza las características principales de las herramientas analizadas.

Característica	Nagios	Zabbix	Prometheus	SolarWinds NPM	PRTG Network Monitor	Sistema Propuesto
Modelo de Monitorización	Monitorización basada en estado de dispositivos	Monitorización de infraestructuras IT	Monitorización basada en series temporales	Monitorización empresarial con análisis de tr	Monitorización basada en sensores	Monitorización web centralizada
Método de Recolección de Datos	Push	Push y agentes	Pull	Push y SNMP	Push y sondas remotas	Push y Pull (según necesidad)
Sistema de Alertas	Sí, configuración manual	Sí, reglas avanzadas	Sí, con Alertmanager	Sí, con AIOps (sin IA avanzada)	Sí, altamente configurable	Sí, alertas configurables por usuarios
Visualización y Dashboards	Interfaz básica con plugins	Dashboards personalizables	Grafana como complemento	Paneles dinámicos personalizables	Visualización integrada	Panel de control propio con personalización
Escalabilidad	Limitada	Alta	Alta (soporta Kubernetes)	Alta, optimizado para grandes redes	Alta, con sondas remotas	Limitada o Alta, dependiendo como se abarque
Integración con APIs REST	Mediante plugins	Integración nativa	Sí, API nativa	Sí, con integraciones empresariales	Sí, mediante API REST	API REST nativa
Compatibilidad con Protocolos de F	SNMP, ICMP, SSH	SNMP, IPMI, JMX, SSH	SNMP, HTTP, gRPC	SNMP, NetFlow, sFlow, WMI	SNMP, WMI, NetFlow, jFlow	SNMP, NetFlow, HTTP, WebSockets
Monitorización Distribuida	No	Sí	Sí	Sí	Sí	Sí
Configuración y Facilidad de Uso	Configuración manual compleja	Interfaz intuitiva, pero requiere configuración inicia	Configuración basada en archivos YAML	Fácil de usar, configuración automatizada	Interfaz intuitiva y sencilla	Configuración simplificada con interfaz intuitiva
Funcionalidades Innovadoras	Extensible mediante plugins, pero poco automatiz	Automatización de tareas de monitorización sin IA	Almacenamiento optimizado de métricas con etiquetado avanzado	Análisis de tráfico con NetPath y PerfStack	Modelo de sensores flexibles para adaptación personalizada	

Figura 4.5: Tabla comparativa con los sistemas de monitorización analizados.

4.7. Conclusiones del estado del arte

Del análisis realizado se desprende que:

- **Nagios** y **Zabbix** son soluciones robustas y muy extendidas, aunque pueden requerir una curva de aprendizaje elevada.
- **Prometheus**, en combinación con Grafana, se ha consolidado como la opción preferida en entornos de microservicios y Kubernetes.
- **SolarWinds** y **PRTG** ofrecen soluciones comerciales completas, con interfaces gráficas avanzadas, pero requieren licencias de pago.
- Todas las herramientas incorporan mecanismos de alertas y notificaciones, siendo este un elemento esencial en la gestión proactiva de redes.

En conclusión, cada herramienta presenta fortalezas y debilidades, lo que evidencia la necesidad de un sistema que combine escalabilidad, facilidad de uso y capacidad de detección de anomalías. Este enfoque será el que guíe el desarrollo del presente TFM.

5. Metodología y Planificación

6. Análisis

7. Diseño

8. Implementación

9. Pruebas

10. Resultados

11. Conclusiones

12. Trabajo Futuro

Bibliografía

AG, P. (2025). *Database monitoring with prtg.* Descargado de https://www.paessler.com/database-monitoring (Último acceso: febrero de 2025)

Ali, S. A. K. (2023). Anomaly detection in telecommunication networks: Towards proactive management. *KUEY Journal*, *30*(5). Descargado de https://kuey.net/index.php/kuey/article/download/3849/2547/8832

Auth0. (2024). *Introduction to json web tokens.* https://jwt.io/introduction/. (Accessed: 2025-08-16)

BetterStack. (2025). *Comparación: Nagios vs zabbix vs prometheus.* Descargado de https://betterstack.com/community/comparisons/nagios-vs-zabbix-vs-prometheus/ (Último acceso: febrero de 2025)

Community, V. (2024). *Vue.js – the progressive javascript framework.* https://vuejs.org/. (Accessed: 2025-08-16)

Developers, C. (2024). *Chart.js documentation.* https://www.chartjs.org/. (Accessed: 2025-08-16)

Edozie, E., Shuaibu, A. N., y Sadiq, B. O. (2025). Artificial intelligence advances in anomaly detection for telecom networks. *Artificial Intelligence Review*. Descargado de https://link.springer.com/article/10.1007/s10462-025-11108-x

Group, P. (2025a). *Prometheus platform data sheet.* Descargado de https://www.prometheusgroup.com/solutions (Último acceso: febrero de 2025)

Group, P. (2025b). *Prometheus solutions overview.* Descargado de https://www.prometheusgroup.com/solutions (Último acceso: febrero de 2025)

Moharam, M. H. (2025). Anomaly detection using machine learning and adopted digital twin concepts in radio environments. *Scientific Reports*, *15*. Descargado de https://www.nature.com/articles/s41598-025-02759-5

Nagios Enterprises, L. (2014). *Nagios resellers datasheet.* Descargado de http://www.nagios.com/products/nagiosxi (Último acceso: febrero de 2025)

Ramírez, S. (2024). *Fastapi documentation.* https://fastapi.tiangolo.com/. (Accessed: 2025-08-16)

Schummer, P. (2024). Machine learning-based network anomaly detection: Design, implementation, and evaluation. *AI*, *5*(4), 2967–2983. Descargado de https://www.mdpi.com/2673-2688/5/4/143

SIA, Z. (2024). *Zabbix datasheet.* Descargado de `https://www.zabbix.com` (Último acceso: febrero de 2025)

SolarWinds. (2025a). *Solarwinds network performance monitor.* Descargado de `https://www.solarwinds.com/network-performance-monitor` (Último acceso: febrero de 2025)

SolarWinds. (2025b). *Solarwinds network performance monitor datasheet.* Descargado de `https://www.solarwinds.com` (Último acceso: febrero de 2025)

Team, B. (2024). *Bootstrap documentation.* `https://getbootstrap.com/`. (Accessed: 2025-08-16)

Technologies, G. (2024). *Gns3 – graphical network simulator.* `https://www.gns3.com/`. (Accessed: 2025-08-16)

Ylonen, T., y Lonvick, C. (2006). *The secure shell (ssh) protocol architecture* (Inf. Téc. no RFC 4251). IETF. Descargado de `https://www.rfc-editor.org/rfc/rfc4251`

A. Anexo I

B. Páginas horizontales