Monitoreo de Infraestructura con Nagios

Juan Manuel Lopez Medina, Juan David Borrero Gómez, Juan Diego Cortes Murillo,
Juan de Dios Rodriguez Perez, Juan Felipe Gómez
Universidad Autónoma de Cali, Colombia

Abstract*—This paper presents the implementation of a monitoring system using Nagios Core to supervise infrastructure within a local network. The solution includes the use of NRPE agents to monitor services such as CPU, memory, disk, and connectivity. A comparison with other tools like Zabbix and Prometheus+Grafana supports the choice of Nagios due to its stability, flexibility, and low resource usage. Results show that Nagios effectively detects incidents in real time and generates alerts to maintain service availability.

Keywords*—Nagios; infrastructure monitoring; NRPE; IT services; open-source; network supervision; availability;

I. INTRODUCCIÓN

En cualquier compañía, la infraestructura tecnológica siempre es el corazón que permite que las operaciones funcionen de manera correcta. Sin embargo, esto no es un impedimento para que los sistemas puedan llegar a fallar inesperadamente; una sobrecarga de un servidor, un disco puede llenarse sin previo aviso, o algún servicio puede simplemente dejar de responder por algún motivo.

Para contrarrestar estos problemas inesperados, el monitoreo de infraestructura se convierte en una necesidad. La implementación de herramientas como Nagios permite supervisar y analizar constantemente el estado de los equipos, los servicios, detectar irregularidades a tiempo, garantizando así, la disponibilidad de los sistemas. Así como todo, el monitoreo de infraestructura tiene sus beneficios, los cuales son de suma importancia para las compañías; los principales son:.

- Detección temprana de problemas: evitando que los fallos menores se conviertan en incidentes graves
- Garantía de disponibilidad: mejorando continuamente la operatividad de los servicios.
- Ahorro de costos: reduciendo pérdidas económicas relevantes para la compañía, debido a interrupciones imprevistas.
- Automatización de alertas: facilitando las notificaciones inmediatas a los operadores antes incidentes.
- Visibilidad total: proporcionando diseños de paneles visuales e históricos de la infraestructura.

II. CONTEXTO

Actualmente, hay muchas compañías que montan su red de equipos y servidores, para lograr gestionar las operaciones diarias. A pesar de esto, no todas cuentan con estrategias adecuadas para vigilar en tiempo real el funcionamiento de su infraestructura tecnológica.

Teniendo una ausencia de la supervisión y vigilancia de esto, se puede llegar a algunas consecuencias, como lo son:

- Caídas inesperadas del sistema y/o la operación.
- Pérdida de información importante.
- Clientes insatisfechos debido a servicios interrumpidos.
- Costos altos en cuanto a recuperación o reparación del sistema.

Por esto y más, se vuelve indispensable el hecho de implementar una solución que permita supervisar continuamente los recursos tecnológicos de la compañía, identificando fallos e inconsistencias antes de que se refleje en el usuario final como una afectación, y tomar las decisiones pertinentes informadas en datos para evitar problemas.

III. ALTERNATIVAS DE SOLUCION

Para conocer más acerca del entorno de monitoreo de infraestructuras, se investigó acerca de herramientas que son conocidas en este ambiente, y se encontró algunas, como:.

- Zabbix: es una plataforma con visualización gráfica, alertas y escalabilidad. Es muy útil en grandes entornos empresariales.
- Prometheus + Grafana: es una solución moderna basada en métricas temporales, con dashboards que son personalizables y una arquitectura adecuada para llevar un monitoreo en la nube.

La siguiente tabla contiene la comparación entre dos de las herramientas de monitoreo: Prometheus + Grafana y Zabbix.:

TABLA I.

COMPARACIÓN ENTRE PROMETHEUS + GRAFANA Y ZABBIX

Características o Aspectos	**Prometheus + Grafana**	**Zabbix**
Uso principal	Recopilación y visualización de métricas temporales de servidores, servicios, contenedores y aplicaciones.	Monitoreo integral de red, servidores, servicios y aplicaciones.
Método de recopilación de datos	Utiliza el modelo pull, obteniendo métricas a través de endpoints HTTP expuestos por los servicios monitoreados.	Utiliza agentes propios (Zabbix Agent) para recopilar datos. También soporta SNMP, IPMI, JMX y monitoreo sin agente.
Visualización de datos	Se apoya en Grafana para representar métricas en dashboards dinámicos, personalizables y en tiempo real.	Representa los datos gráficamente mediante mapas, gráficos y tableros de control propios.
Disponibilidad del servicio	No incluye funciones nativas para reportes de disponibilidad del servicio; puede integrarse con herramientas externas para ello.	Incluye una función integrada para generar informes de disponibilidad, incluyendo planificación de interrupciones.
Escalabilidad	Altamente escalable, ideal para entornos en la nube y arquitecturas de microservicios.	Escalable, aunque puede requerir ajustes adicionales en grandes infraestructuras.
Curva de aprendizaje	Requiere conocimientos previos de métricas, PromQL y configuración de dashboards en Grafana.	Más amigable para principiantes, con una interfaz web centralizada para gestionar casi todo.
Alertas y notificaciones	Permite configurar alertas mediante Alert Manager, integrable con múltiples canales (email, Slack, etc.).	Sistema de alertas potente integrado directamente en el servidor de monitoreo.
Licenciamiento y costo	Gratuito y de código abierto (tanto Prometheus como Grafana son proyectos bajo licencia Apache 2.0).	Gratuito y open-source. Zabbix ofrece versiones comerciales (Enterprise) con soporte profesional y características avanzadas.

Ambas herramientas son buenas y pueden ser tomadas como opciones de implementación, sin embargo, en este caso se opta por Nagios, debido a su estabilidad, comunidad activa, su posible extensión mediante plugin, y su enfoque que clásico y orientado a la supervisión del estado de los servicios y los equipos en las redes locales.

(1)

IV. DISEÑO DE LA SOLUCIÓN

A. RED MONTADA

Se diseñó una red local simuladas con el rango de direcciones IP 192.168.50.0/24, usando Vagrant y VirtualBox para visualizar los distintos equipos; la red incluye:

1) Servidor central con Nagios.

2) Varias máquinas virtuales simulando servidores, clientes y firewalls.

B. HERRAMIENTAS DE MONITOREO

Se instaló Nagios Core 4.5.1 como sistema centralizado de monitoreo, permitiendo la herramienta:.

1) Comprobar la disponibilidad de los equipos.

2) Verificar el estado de los servicios críticos, como, CPU, RAM, disco, y demás.

3) Visualizar alertas y cambios en tiempo real mediante el dashboard web proporcionado.

C. COMUNICACION ENTRE NAGIOS Y LOS EQUIPOS

Para lograr esto, se utilizó el plugin NRPE (Nagios Remote Plugin Executor), el cual permite la comunicación entre Nagios y los nodos monitorizados. Con este plugin, Nagios puede ejecutar comandos de verificación en equipos remotos para conocer su estado en tiempo real.

D. DASHBOARDS Y VISUALIZACIÓN

Nagios proporciona un dashboard web el cual muestra el estado de cada equipo y servicio.

- Alarmas en caso de sobreuso de recursos.
- Historial de eventos.
- Vista detallada por equipo y por métrica monitoreada.

V. IMPLREMNTACION

1) PREPARACIÓN DEL ENTORNO.

- Creación de máquinas virtuales con Vagrant y VirtualBox.
- Configuración de una red local en el rango 192.168.50.0/24.
- Despliegue de:
- ServidorNagios.
- Equipos virtuales a monitorear (servidor, cliente adicional, firewall).

2) INSTALACIÓN DE NAGIOS CORE.

- Instalación manual de Nagios Core 4.5.1 en el servidor central.
- Configuración del acceso web para el dashboard de monitoreo.
- Validación de la configuración inicial.

3) INSTALACIÓN NRPE Y PLUGINS.

- En los equipos monitorizados se instaló el agente de NRPE (NRPE Agent).
- Se añaden plugins para monitoreo del uso de CPU, uso de memoria RAM, espacio en disco, y demás.

4) INSTALACIÓN DE NAGIOS CORE.

- Se definen los hosts y servicios en los archivos de la configuración
- Se configuró algunos de los siguientes servicios:
- Ping para verificar la conectividad.
- Check_load para la carga del procesador.
- Check_mem para uso de memoria.
- Verificación del estado de servicios importantes.

VI. PRUEBAS

Las pruebas se diseñaron para validar la efectividad del monitoreo y la capacidad de Nagios para reaccionar ante situaciones irregulares:

VERIFICACIÓN DEL DASHBOARD

- Se accedió a la interfaz web de Nagios y se visualizó el estado de los equipos y servicios configurados.
- El estado se mostraba con indicadores visuales de colores, OK con verde, WARNING con amarillo y CRITICAL con rojo.

PRUEBA DE CAMBIO EN TIEMPO REAL

- Se apagó una de las máquinas virtuales para simular una caída.
- Nagios detectó la desconexión en menos de 5 minutos.
- Se generó una alerta crítica de manera automática en el dashboard.

PRUEBA DE CARGA

- Se aumentó intencionalmente la carga del CPU y se redujo la memoria disponible.
- Nagios reportó correctamente el estado crítico en los servicios que fueron afectados.

VII. DISCUSIÓN DE LAS PRUEBAS

CAPACIDAD DE DETECCIÓN EN TIEMPO REAL

- Las pruebas realizadas demostraron que Nagios tiene una excelente capacidad de detectar inconsistencias en tiempo real. Las caídas de una máquina o la saturación de un recurso fueron identificadas de manera rápida, lo cual es necesario para entornos que demandan alta disponibilidad.

DASHBOARD INTUITIVO Y ÚTIL

- El panel proporciona una visualización clara, detallando cada equipo y métrica, el uso de colores facilita la identificación de forma inmediata de lo que está ocurriendo en el momento del monitoreo, lo cual es algo más ágil para el operador o administrador que se encuentre en el área, y así tomar las medidas pertinentes.

SIMULACIÓN DE ESCENARIOS CRÍTICOS

- Los escenarios probados en el testeo, fueron la pérdida de conectividad, saturación de recursos, fallos en los servicios y, en todos los casos, Nagios respondió adecuadamente para cada una de estas inconsistencias.

FLEXIBILIDAD Y ESCALABILIDAD

- Se pudo obtener que Nagios es fácilmente extensible mediante los correctos plugins, permitiendo añadir métricas y servicios a monitorear. Aunque, se identificó algunos retos, como lo son el tema de configurar adecuadamente los archivos .cfg, y la instalación manual de ciertos plugins, que requiere de conocimientos técnicos adicionales a lo tradicional.

VIII. CONCLUSIONES

- El monitoreo de infraestructura es esencial para la supervisión en tiempo real que permite anticiparse a fallos inesperados, minimizar tiempos de inactividad y optimizar el rendimiento de los sistemas monitoreados.
- Nagios es una solución robusta y confiable, la cual permite un monitoreo detallado de múltiples aspectos de la red, con una interfaz clara y funcionalidades que son suficientes para entornos empresariales medianos.
- El aprendizaje técnico es de suma importancia, ya que la implementación de la solución fortaleció habilidades que son claves para la administración de redes, configuración de servicios, monitoreo remoto y el análisis de datos operativos.
- La automatización mejora la gestión de TI, aplicando las alertas automáticas basadas en los datos y lo que acontezca, la visualización en tiempo real, permitiendo tomar decisiones informadas, lo cual hace que mejore la eficiencia operativa y la experiencia del usuario final, y llegar a implementar correctamente proyectos a futuro para la compañía en donde se aplique estas herramientas.

REFERENCIAS

[1] Nagios Enterprises, "Nagios Core Documentation," Nagios.org, 2024. [Online]. Available: https://www.nagios.org/documentation

[2] Debian Wiki, "Nagios NRPE," wiki.debian.org, 2023. [Online]. Available: https://wiki.debian.org/nagios-nrpe

[3] M. Sharma, "Introduction to Grafana, Prometheus, and Zabbix," DZone, 2021. [Online]. Available: https://dzone.com/articles/introduction-to-grafana-prometheus-and-zabbix