

# El ecosistema de Nagios.

Bienvenidos al Circo de Nagios. Vamos a tratar de aclarar en estas páginas el panorama actual de herramientas, clientes, plugins, addons,... que existen alrededor de Nagios Core.

Nagios es un sistema de monitorización con una lógica muy potente debajo que nos permite monitorizar prácticamente todo lo que tenga bits, pero por si solo tiene muchas carencias que lo hacen una herramienta hosca, cerrada y poco manejable. Para solucionar estos problemas distintos desarrolladores han ido creando y manteniendo herramientas que cubren distintas necesidades que no proporciona Nagios o lo hace de forma somera o poco práctica.

Nos centraremos en Nagios Core, ya que existe una versión Nagios de pago (pago por soporte y actualizaciones) llamada Nagios XI que, también basada en Nagios Core y creada por el desarrollador de este, intenta solucionar sus carencias (y amortizarlas, es lógico querer vivir de tu trabajo).

## **Plugins.**

Los “oficiales” son los plugins estándar para monitorización, que no proporciona Nagios si no que si distribuyen aparte (si, Nagios no proporciona plugins para monitorizar, son aparte, tiene su lógica). Además de los plugins estándar de monitorización existen infinidad de plugins para chequear distintos aspectos de sistemas y software proporcionados por terceros. La mayoría de ellos agrupados y clasificados en una página concreta aunque otros muy importantes los podamos solo encontrar en las páginas de sus desarrolladores o patrocinadores.

## **Chequeos locales.**

Los plugins estándar nos proporcionan chequeos o bien de red (desde otro equipo) o bien, si son locales, es necesario que estén presentes en el equipo a chequear y usar una forma de “activarlos” por red (chequeos activos) o programar su ejecución y que reporten por red (chequeos pasivos). Para esos chequeos en local en los equipos disponemos de varias herramientas y/o clientes. Desde una simple conexión desde el host Nagios mediante ssh para su ejecución y recogida de datos hasta clientes instalados localmente en el equipo que nos facilitan y/o integran los chequeos y la configuración de estos.

## **Configuradores.**

La configuración de chequeos y todos los objetos relacionados con estos se realiza mediante ficheros de texto. La curva de entrada para entender y controlar el uso de estos es cierto que es un tanto elevada pero una vez entiendes sus relaciones y propiedades y sabes usar la “herencia” de objetos y plantillas tienes toda la potencia del copy / paste a tu disposición :-). Nagios Core no proporciona un interface amigable para la creación y gestión de nuestros objetos pero existen aplicaciones que lo hacen y muy bien. Ambas se basan en una pequeña BBDD donde guardan las configuraciones que luego mueven a ficheros estándar de configuración de Nagios.

## ***Interface de visualización.***

El interface de visualización de Nagios es bastante espartano y poco modificable, básicamente la única forma que tienes de modificarlo es cambiarle las hojas de estilo y añadir cabeceras y pies de página, poco más. Debemos tener en cuenta que ya es mayorcito, está escrito en C y son CGIs compilados los que nos muestran los datos en pantalla. Sí, es lo más rápido probablemente, pero para su modificación requiere saber mucho C y andar compilando cada CGI que se modifique. Para suavizar tanta austeridad existen interfaces adicionales creados por terceros, desde los que aportan básicamente hojas de estilo más modernas y vistosas hasta los que reescriben completamente el interfaz normalmente en otro lenguaje más versátil.

## ***Más vistosidad y funcionalidad.***

Para hacer aún más visual el control del colega existen herramientas también de terceros que nos proporcionan gráficos a partir de los datos obtenidos de los chequeos o que nos dejan situar el estado de los chequeos online sobre un mapa o dibujo de nuestra cosecha. Piensa lo chulo que puede quedar ese Visio que te curraste con lucecitas verdes, amarillas o rojas según el estado de tus equipos / servicios, y... ¡en tiempo real!

## ***Acceso a datos.***

Muchas de estas herramientas tienen que acceder a los datos de Nagios, tema delicado ya que la forma estándar de almacenar los datos de los chequeos que tiene Nagios no es muy eficiente. Para solucionar este tema de nuevo existen varias herramientas que nos permiten, desde duplicar los chequeos y/o el estado de los objetos en pequeñas BBDD hasta la más novedosa y funcional que nos permiten preguntar mediante una aproximación SQL sobre el estado de los objetos. Estas herramientas son muy importantes ya que muchas otras se apoyan en estas para recoger los datos, de hecho normalmente soportan de forma configurable el uso de más de una de estas herramientas.

## ***Simplificación.***

Para facilitar el uso e instalación inicial de Nagios Core existen desde máquinas virtuales ya preconfiguradas con muchas de estas funcionalidades hasta un empaquetado con muchas de ellas ya incluidas que nos facilita su instalación y actualización de un plumazo. Dicho empaquetado realizado por terceros nos facilita la instalación (y más cosas) de prácticamente las pistas más importantes del circo de Nagios de una tacada.

## ***Sofisticación.***

Un asunto que echa de menos en Nagios es poder monitorizar de forma más próxima a la lógica de

negocio. Es decir, tenemos mucha potencia para monitorizar el estado de equipos y sus servicios pero estos resultados nos quedan muy lejos de la interpretación de cómo afectan a mi negocio. Si tengo numerosas máquinas en cluster / HA para servir una aplicación necesito algo más que saber que se ha caído un servicio de una máquina. Necesito saber si afecta o no y como a mi Aplicación dicho error. Al fin y al cabo tengo una arquitectura de alta disponibilidad para soportar esos pequeños fallos. ¿No estaría bien poder definir de forma lógica la relaciones entre los nodos / servicios que forman parte de tu aplicación y que Nagios te alertará en función de dicha lógica? Suena bien verdad. También podemos hacerlo con otro maravilloso “addon” para Nagios.

## **Forks.**

Como no, cada herramienta Open Source que se precie tiene al menos un fork. En este caso también tenemos una aplicación basada originalmente en Nagios Core, que ahora es “competencia” de esta y que también nos interesa conocer porque puede aportarnos cosas interesantes que nos puedan hacer decantarnos por ella. Además, todas las herramientas de las que hablamos también funcionarán con este...

Efectivamente, no hay enlaces en este post. Es para que no te me despistes. Ya iremos viendo cada una de las aplicaciones / utilidades en distintos posts. Si hubiera puesto enlaces ya estarías dispersándote con un montón de pestañas abiertas :-).