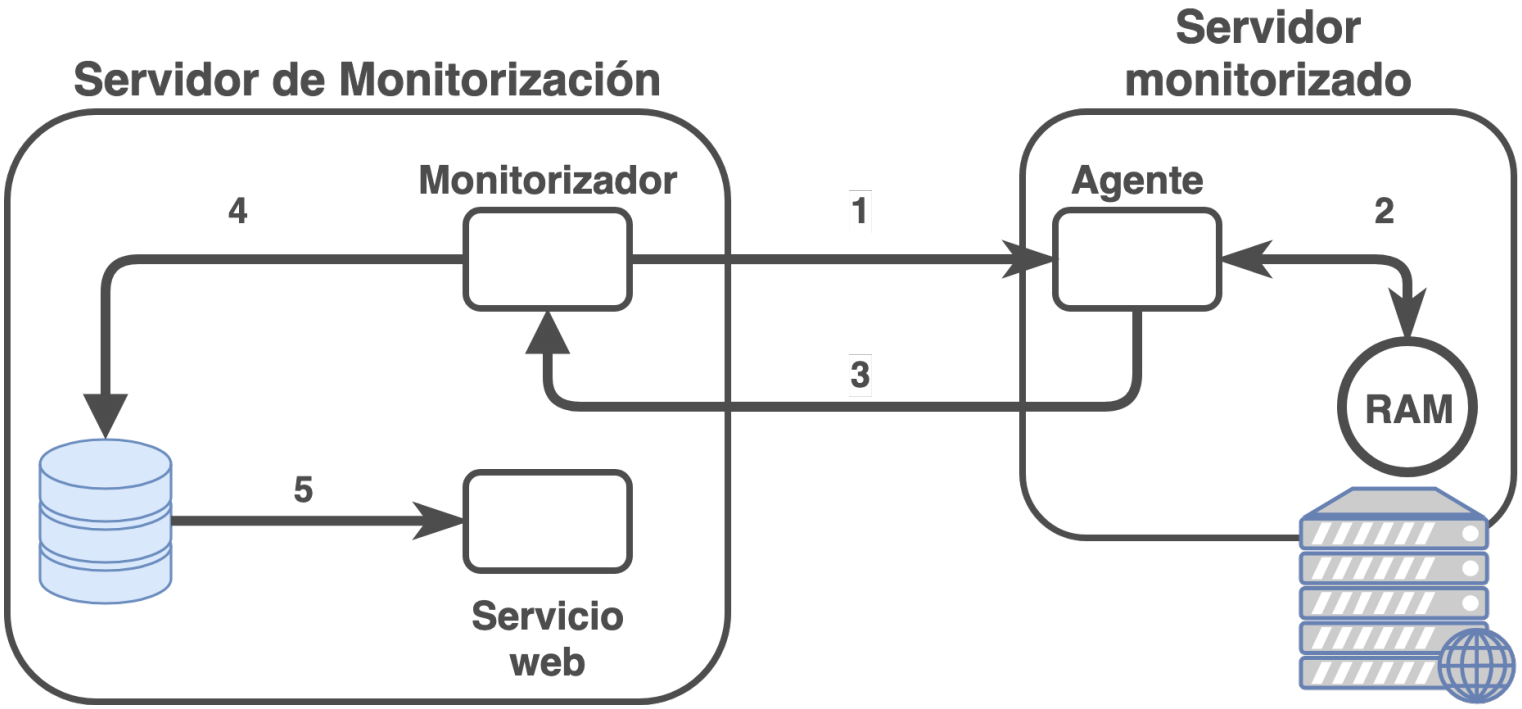


Sistemas de monitorización

Rubén Gómez Olivencia

2021-2022

- Github: https://github.com/yuki

Puedes encontrar la última versión de este libro en formato **HTML** en el siguiente link, así como otros libros que he creado. Para descargar el código fuente en formato **Markdown** visita el repositorio en GitHub.

Información

Por favor, ponte en contacto conmigo si encuentras algún fallo, falta de ortografía o quieres mejorar de alguna manera este libro. Gracias.

1. Sistemas de Monitorización

El sistema de monitorización se encarga de recopilar información acerca del estado de los servidores de una infraestructura. Entre las métricas y datos que debe recopilar se encuentran:

- **Estado del servidor**: cantidad de RAM utilizada, estado de los discos duros, carga del servidor, sistema de ficheros, …
- Estado de los **servicios que tiene el servidor**: servidor web, número de procesos que tiene, bases de datos, conexiones existentes, cola de correos electrónicos para enviar si es un servidor de correo, … Dependiendo del servicio habrá que realizar unas comprobaciones u otras.
- **Infraestructura en la que se encuentran**: estado de la red, conexión a otros servidores, …
- **Estado del clúster**: en caso de que el servidor pertenezca a un clúster, hay que comprobar que el clúster se encuentra en perfecto estado.
- **Dependencias externas**: un servidor puede depender a su vez de otros, o de servicios externos, que deben de estar funcionando de manera correcta.

Normalmente en la monitorización actúa un **agente** (o servicio) instalado en el equipo monitorizado que obtiene la información requerida que se envía a un servidor central que recopila la información de toda la infraestructura monitorizada.

Esta información suele ser almacenada durante un periodo de tiempo determinado (un año, por ejemplo) para poder ser usada y comparar la situación de los servidores a lo largo del tiempo. Gracias a esta comparación temporal **se puede llegar a predecir el estado del servidor** a unos días/semanas vista y evitar problemas antes de que sucedan (no tener espacio en discos duros, mal funcionamiento de servicios por falta de RAM, …).

Información

La monitorización de servicios y equipos dentro de una infraestructura debe considerarse parte del proyecto, ya que es una parte muy importante de cara al mantenimiento del mismo.

1.1. Monitorización de servidores

Es habitual que los sistemas de monitorización funcionen en base a plantillas, que posteriormente se pueden asociar a los servidores monitorizados. Estas plantillas contendrán los servicios que deben ser monitorizados en cada tipo de servidor, ya que no es lo mismo monitorizar un servidor web o un servidor con un SGBD.

Para monitorizar un servidor lo habitual suele ser realizar las siguientes operaciones:

- **Comprobar conectividad con el servidor**: suele ser habitual que los servidores estén fuera de nuestra red (en un proveedor de Internet, en un cliente, en otra oficina…), por lo que es necesario

que exista conectividad de alguna manera para poder realizar la monitorización. En caso de no estar en nuestra red, el uso de una VPN es lo más utilizado.

- **Instalar un agente en el propio servidor**: Será el encargado de recopilar la información necesaria para mandarla al servidor central. Dependiendo del sistema de monitorización utilizado, necesitaremos un tipo de agente u otro. Algunos se encargan de realizar todas las comprobaciones y otros llaman a otros programas para realizar las comprobaciones y después mandar el resultado al servidor central.
- **Dar de alta el servidor en el sistema centralizado**: Tal como se ha dicho previamente, lo habitual es contar con un sistema centralizado en el que se tendrán todos los servidores y el estado de las comprobaciones realizadas. De ser así, habrá que darlo de alta, y para ello se necesitará:
  - **Nombre del servidor**: Un nombre que a simple vista identifique el servidor. Suele ser habitual poner el nombre del cliente también, y/o el tipo de servicio que preste.
  - **IP del servidor**: Para poder realizar la conexión al servidor.
  - **Plantillas asociadas**: En caso de utilizar un sistema que utilice plantillas, al dar de alta el servidor se le aplicarán las plantillas necesarias para que realicen todos los checks oportunos. Por ejemplo: plantilla de Servidor Linux + plantilla de Servidor web + Plantilla de MySQL.
  - **Puerto de conexión**: Los agentes de monitorización suelen contar con un puerto que queda a la escucha. Si hemos cambiado el puerto, habrá que indicarlo a la hora de dar de alta.
  - **Otras opciones**: Dependiendo del sistema de monitorización se podrán añadir muchas más opciones, como por ejemplo:
    - **Servidores de los que se depende**: Imaginemos que el servidor monitorizado depende a su vez de un router que también está monitorizado. Si el router cae, no llegaríamos al servidor, por lo que realmente es una caída por dependencia, aunque el servidor puede estar funcionando de manera correcta. Esto nos puede permitir crear “árboles de dependencias” de servidores.
    - **Periodos de monitorización**: Lo habitual es que un servidor esté monitorizado 24x7, pero quizá nos interese realizar cambios y que sólo se monitorice en unas horas determinadas (quizá el resto del tiempo está apagado).
    - …

1.2. Funcionamiento de la monitorización

Para conocer cómo funciona un sistema de monitorización lo mejor es que tomemos como ejemplo un tipo de servicio que queremos monitorizar. Como ejemplo se puede tomar las comprobaciones que queremos realizar a un SGBD (Sistema Gestor de Bases de Datos).

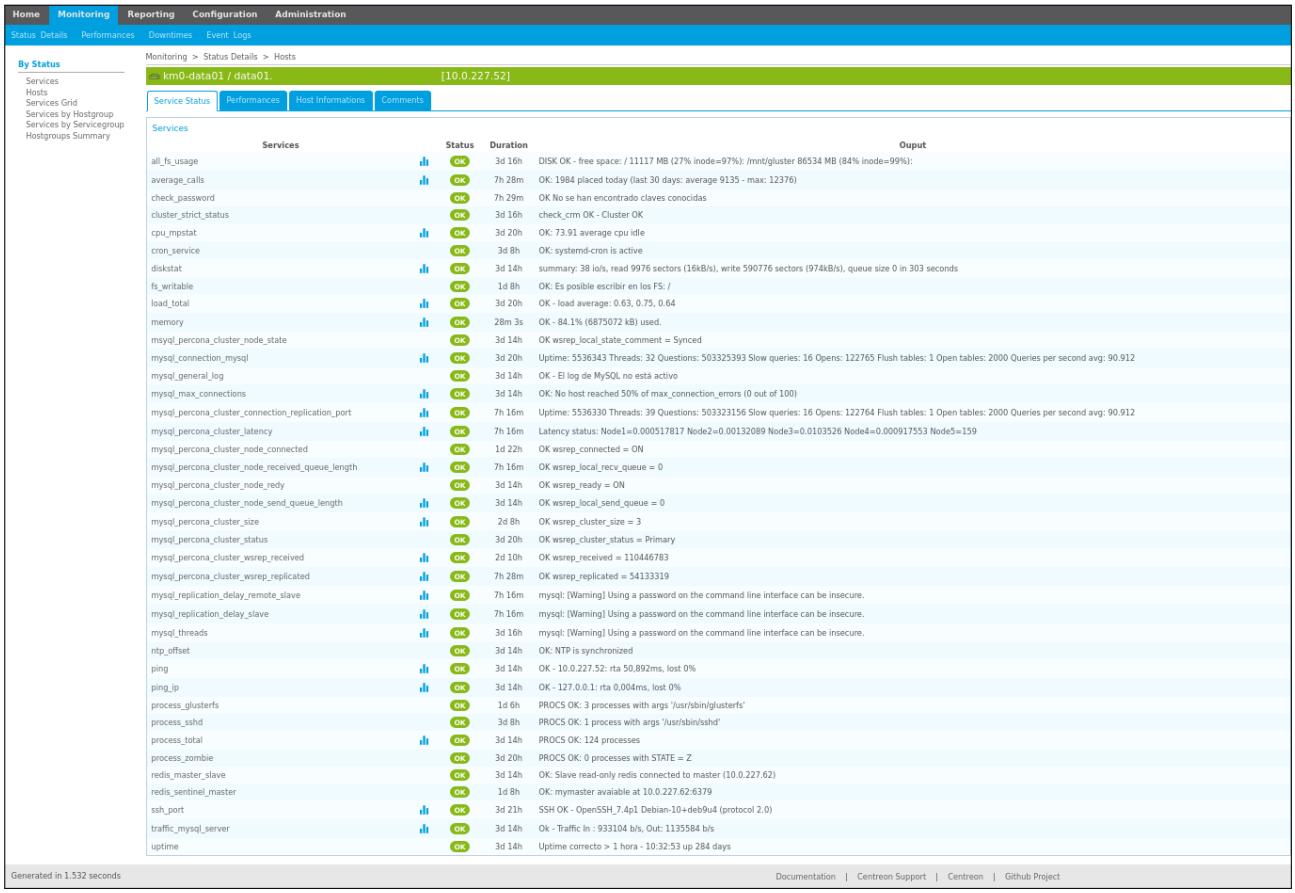
No será lo mismo realizar la monitorización de un servidor MySQL o de un Oracle, pero las comprobaciones

que queremos realizar en ellos deberían ser similares. Vamos a querer realizar la monitorización de las mismas comprobaciones: estado de las tablas en memoria, número de hilos en ejecución, número de *slow_queries*, …; pero los scripts ejecutados serán distintos.

Información

La monitorización dependerá del propio servicio que vayamos a monitorizar.

A continuación se puede ver el estado de un servidor monitorizado a través del sistema de monitorización Centreon:



Servidor monitorizado en Centreon

En la imagen anterior se puede comprobar un número de ***checks***, o comprobaciones, que se están realizando sobre un servidor concreto. Cada fila es una comprobación y contienen:

- **Nombre del check/servicio**: Un nombre para identificar qué es lo que se está comprobando con el check.
- **Icono para mostrar gráficas**: Algunos checks recibirán información que puede ser graficada para así poder observar patrones en el comportamiento del servidor. Por ejemplo: cantidad de RAM ocupada, número de procesos en el sistema, número de conexiones a un servidor, …
- **Estado del check**: Normalmente, tras realizar la comprobación, el check termina con uno de los

siguientes resultados:

- **OK**: El resultado obtenido es el correcto.
- **Warning**: El resultado obtenido está entre los márgenes de peligro. Es posible que de seguir así pase al estado siguiente:
- **Crítico**: El servicio devuelve un estado que es considerado crítico, lo que puede hacer que llegue a mal funcionamiento del mismo, o incluso que el servidor comience a dejar de funcionar (imaginemos que el servidor está con el 90 % de la RAM ocupada o de disco duro ocupado).
- **Indeterminado**: Por alguna razón, el *check* no se ha realizado, o el valor devuelto es indeterminado o no se puede saber en qué otro estado situarlo.

- **Duración del estado**: Para conocer cuánto tiempo lleva en el estado la comprobación obtenida. Lo ideal es que nunca haya estados que no sean OK y por lo tanto la duración de los mismos sea lo más alta posible.
- **Valor devuelto por la monitorización**: El valor real devuelto por la comprobación realizada. En base a este resultado se puede realizar las gráficas mencionadas previamente.

Información

El estado del servicio dependerá del valor devuelto por la monitorización.

Este resultado se cotejará con los valores que hayamos puesto para que sea considerado OK, Warning o Critical. Es decir, **en algunos casos el estado del servidor depende de los valores devueltos y de la baremación que le hayamos otorgado**.

Pongamos como ejemplo la monitorización de un SGBD:

- **El servicio del SGBD está funcionando**: Ahí no hay baremación posible. Si el servicio no está arrancado, es lógico pensar que el estado es crítico y que por tanto hay que ver qué ha ocurrido.
- **Número de conexiones en el SGBD**: El resultado devuelto será un número entero (que podremos graficar para obtener patrones). En este caso, podemos decidir los rangos para que el resultado sea OK, Warning o Critical. Es decir, si el resultado obtenido está por debajo del umbral de Warning, el sistema considerará que el estado es OK. Si está en dicho rango, será Warning y si está en el rango de Critical, así lo indicará.

Esta baremación y **estos rangos** se suelen aplicar también en las plantillas de los servicios. Hay que entender que también **pueden ser modificados y personalizados para un servidor concreto**. No es lo mismo que un SGBD tenga 500 conexiones simultáneas si tiene 8Gb de RAM o si tiene 128Gb (en el primer servidor se puede considerar que es un estado crítico mientras que en el segundo es lo esperado).

Cuando un *check* termina siendo un Warning o un Critical **es habitual que haya un sistema de alarmas configurado**. Dependiendo del sistema utilizado, notificará a los administradores mediante e-mail,

mensajería instantánea, SMS, … para que realicen un análisis lo antes posible y solucionen el estado del servicio.

Información

Los sistemas de monitorización suelen contar con un sistema de alarmas para que nos avise de los servicios caídos.

1.2.1. Monitorización básica

Tal como se ha comentado, en los servidores se suele realizar una monitorización del estado del mismo que suele ser común para todos, por lo que lo habitual suele ser tener una plantilla genérica para todos los servidores con la que se monitorizará:

- Cantidad de RAM utilizada
- Cantidad de memoria virtual utilizada
- Carga de la CPU
- Espacio libre en las unidades de disco duro
- Estado del sistema RAID del servidor (en caso de tenerlo)
- Cantidad de usuarios conectados a la máquina
- Estado de puertos de conexión (SSH, por ejemplo)
- Latencia hasta llegar al servidor
- …

Es cierto que no será lo mismo monitorizar un sistema GNU/Linux o un sistema Windows (ya que puede variar alguno de las comprobaciones a realizar), pero el estado general que queremos conocer es el mismo. Por lo tanto, lo habitual es tener dos plantillas, una específica para servidores Windows y otra para GNU/Linux.

1.2.2. Monitorización de Servicios

Aparte de la monitorización básica comentada previamente, necesitaremos monitorizar el estado de los servicios que pueda tener el servidor propiamente dicho. Para ello, de nuevo, se crearía una plantilla específica para cada tipo de Servicio que podamos tener en nuestro servidor.

No será lo mismo monitorizar un servidor que tenga un servidor web, un servidor de base de datos, un proxy… O puede que el servidor cuente con todos esos servicios.

Es por eso que a la hora de realizar la monitorización de un servidor **es muy importante conocer qué funciones desempeña cada servidor en la infraestructura a la que pertenece** y analizar los servicios que tiene arrancados para posteriormente ser monitorizados.

Información

Es muy importante conocer qué funciones desempeña cada servidor en la infraestructura a la que pertenece.

1.3. Tipos de monitorización

Existen varias maneras de realizar la monitorización de un servidor, y dependerá del gestor de monitorización que usemos (en caso de usar uno).

Es habitual que cuando nos referimos a sistemas de monitorización lo dividamos en dos grandes familias:

- Monitorización Activa
- Monitorización Pasiva

Estas dos maneras de monitorización suelen ser excluyentes, aunque algunos sistemas de monitorización permiten ambas, por lo que nos puede interesar usar una u otra dependiendo de la situación.

1.3.1. Monitorización pasiva

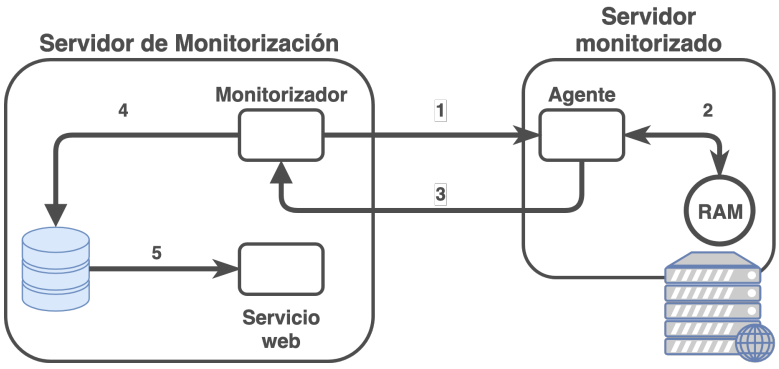
En la monitorización pasiva el servidor (u objeto monitorizado) es el encargado de mandar la información de manera periódica al servidor central. El agente instalado se ejecutará como una tarea programada cada cierto tiempo (habitualmente unos pocos minutos) e informará de la situación cambiante, de haberla, al servidor central.

Esta manera de monitorización es utilizada también cuando no hay un servidor central. En este caso, si la comprobación ha sido incorrecta, podría mandar un mail al administrador del servidor.

1.3.2. Monitorización activa

Suele ser la manera habitual de proceder de los sistemas que cuentan con un servidor centralizado de monitorización. El servidor de monitorización se encarga de preguntar al servidor, a través de la conexión con el agente, por la comprobación de alguno de los checks, y el agente devuelve la información.

A continuación se puede observar las etapas que existen en un sistema de monitorización activa utilizando un servidor de monitorización central:



Proceso de monitorización activa

Las etapas serían:

0. El sistema de monitorización tiene un scheduler (o planificador) que decide cuándo tiene que realizar cada comprobación (normalmente, cada pocos minutos).

1. El servicio encargado de monitorizar **establece conexión con el agente remoto** y le pide que compruebe un estado. En este ejemplo se ha optado por la RAM.

2. El agente en el servidor que se quiere monitorizar recibe la notificación y realiza una **comprobación local** (normalmente llamando a scripts locales) para obtener la cantidad de RAM ocupada, total y libre que tiene

3. El agente envía al sistema de monitorización el resultado obtenido en la ejecución de los scripts del paso anterior.
    1. El monitorizador al recibir el resultado, lo coteja con los rangos de baremación que tiene y decide si el check está en estado OK, Warning o Critical.

    2. Lo habitual es que si el resultado del servicio no es OK, se ejecute en el servidor de monitorización algún tipo de alarma (ya sea enviar un mail, sistema de mensajería, …) para notificar a los administradores.

4. El sistema de monitorización guarda en una base de datos los resultados obtenidos para así poder realizar posteriores análisis o comprobaciones temporales de los mismos.

5. Esos datos se suelen visualizar en una interfaz web, tal como hemos visto previamente.

Estos pasos son ejecutados de manera continuada en el servidor de monitorización para cada comprobación que se realiza en cada uno de todos los servidores que se monitorizan. Por lo tanto, se entiende que el propio servidor de monitorización también tiene que ser monitorizado ya que es de vital importancia que su estado sea óptimo.

1.3.3. Monitorización centralizada

Como ya se ha comentado, es el sistema habitual de monitorización. Las ventajas que podemos obtener al hacer uso de este sistema son muchas, pero se pueden destacar las siguientes:

- **Monitorización centralizada**: Aunque parezca obvio, el tener un único sistema en el que concentrar toda la información es muy útil y eficaz.
    - La alternativa sería tener una monitorización distinta en cada servidor.

- **Interfaz web**: Hoy en día suele ser habitual que los sistemas de monitorización tengan un servicio web en el que visualizar todos los datos obtenidos.

- **Sistema de plantillas**: De nuevo, es lo habitual, lo que hace que la gestión de monitorización de servidores sea más cómoda.

- **Gestión de usuarios**: Podremos tener usuarios que puedan ver unos servidores u otros, por lo que podemos tener equipos especializados en distintos grupos de monitorización y que sólo se enfoquen en ellos.

- Esto también es útil para dar acceso a los clientes a la monitorización de sus propios servidores.

1.3.4. Monitorización reactiva

La monitorización reactiva se puede definir como el sistema de monitorización que no sólo se encarga de comprobar y recibir el estado de los servidores, si no que también reacciona a los mismos para tratar de solucionar los problemas encontrados. Tras esta definición está la idea de que **existen ciertos fallos recurrentes que no siempre necesitan la intervención humana para solucionarse**, y que por tanto, se puede tratar de ejecutar antes de que sea considerado un problema real.

Como **ejemplo sencillo** se puede poner **el espacio libre en disco duro**. Imaginemos que se comprueba que apenas hay espacio en el disco duro de un servidor. En este caso, el sistema de monitorización recibirá que el servidor **está al 99.95 %** de espacio ocupado, y por tanto, en lugar de notificar a un humano indicando el estado crítico, **el sistema reacciona de manera automática tratando de liberar espacio**. Se habrá configurado previamente que en la reacción de este error trate de borrar ficheros temporales, vaciar papelera, limpiar ficheros de caché de ciertas rutas … Una vez hecho esto, se volverá a comprobar el estado del servidor. Si el espacio ocupado en disco duro ha bajado y está en modo OK no habrá que hacer nada más, y se habrá evitado que un administrador tenga que realizar dicha tarea. Si por el contrario el estado sigue siendo incorrecto, el sistema notificará el error para que se realice un análisis y se solucione el problema.

Como **ejemplo extremo** (que no suele ser habitual configurarlo así), imaginemos que **la RAM consumida por un SGBD es muy alta** y esté poniendo en peligro el estado del servidor, se podría configurar para que **el sistema reaccione reiniciando el SGBD para que libere la RAM** y vuelva a prestar servicio.

1.4. Gestores de monitorización

Hoy día existen muchos sistemas de monitorización, y dependiendo de nuestras necesidades deberemos optar por uno u otro. A continuación se expondrán varios ejemplos de gestores de monitorización basados en Software Libre, aunque la gran mayoría de ellos cuentan con un sistema dual. Es decir, se puede descargar y montarlo en tu propio servidor o puedes contratar a la empresa para que ellos tengan el servicio central:

- **Nagios**: Se puede considerar uno de los sistemas de monitorización más conocidos y del que se han basado otros. Generó mucha comunidad de administradores creando muchos scripts/plugins para hacerlos funcionar con él. Estos mismos scripts suelen ser utilizables en otros sistemas de monitorización.
- **Centreon**: Originalmente se creó como interfaz web para Nagios, pero poco a poco fue sustituyendo partes de Nagios hasta terminar siendo un sistema de monitorización completo. Existe la posibilidad de realizar la instalación por paquetes, descargar el sistema operativo en una ISO que te instala todo o incluso una máquina virtual con todo ya instalado y con configuración básica. (Demo).
- **PandoraFMS**: Sistema de monitorización creado por el español Sancho Lerena Urrea. Al igual que los anteriores, tiene sistema dual y la instalación se puede realizar por varios métodos.
- **Cacti**: Sistema más sencillo que los anteriores y habitualmente utilizado sólo en servidores sueltos,

es decir, no de manera centralizada.

- **Munin**: Igual que el anterior, ideal para monitorizar unos pocos servidores, ya que no se puede considerar un sistema centralizado como los primeros. Ver anexo de instalación de Munin.

Existen otros sistemas de monitorización basados “en la nube”, cuya funcionalidad es similar a lo expuesto previamente. Para hacer uso de estos sistemas nos descargamos un agente, lo instalamos y se encargará de mandar la información a los servidores de la plataforma contratada. Lógicamente, dependiendo del gasto realizado obtendremos más o menos servicios. Entre este tipo de servicios se pueden destacar:

- New Relic
- DataDog