ESCUELA TÉCNICA SUPERIOR DE INGENIEROS INDUSTRIALES Y DE TELECOMUNICACIÓN

UNIVERSIDAD DE CANTABRIA

Trabajo Fin de Grado

Estudio e implementación de un entorno de gestión para la red privada del laboratorio docente de telemática (GIT-UNICAN)

(Study and implementation of a management environment for the private network of the telematics teaching laboratory (GIT-UNICAN))

Para acceder al Título de

Graduado en Ingeniería de Tecnologías de Telecomunicación

Autor: Alberto Fernández Añivarro

Julio – 2018

E.T.S. DE INGENIEROS INDUSTRIALES Y DE TELECOMUNICACIÓN

GRADUADO EN INGENIERÍA DE TECNOLOGÍAS DE TELECOMUNICACIÓN

CALIFICACIÓN DEL TRABAJO FIN DE GRADO

Realizado por: Alberto Fernández Añivarro

Director del TFG: José Angel Irastorza Teja

Título: Estudio e implementación de un entorno de gestión para la red privada del laboratorio docente de telemática (GIT-UNICAN)

Title: Study and implementation of a management environment for the private network of the telematics teaching laboratory (GIT-UNICAN)

Presentado a examen el día:

Para acceder al Título de: GRADUADO EN INGENIERÍA DE TECNOLOGÍAS DE TELECOMUNICACIÓN

Composición del Tribunal:
Presidente (Apellidos, Nombre): José María Zamanillo Sainz de la Maza
Secretario (Apellidos, Nombre): Alberto Eloy García Gutiérrez
Vocal (Apellidos, Nombre): José Angel Irastorza Teja

Este Tribunal ha resuelto otorgar la calificación de: .....

Fdo.: El Presidente **Fdo.:** El Secretario

Fdo.: El Vocal **Fdo.:** El Director del TFG (sólo si es distinto del secretario)

Vº Bº del Subdirector Trabajo Fin de Grado Nº (a asignar por Secretaría)

Agradecimientos

En primer lugar, he de agradecer la ayuda, atención y orientación que mi tutor, José Angel, me ha prestado durante estos meses de trabajo, sin el cual no hubiera podido llevar a cabo este proyecto.

He de dar las gracias también a mis compañeros Raúl, Jorge y Fernando por el apoyo mutuo que nos hemos prestado durante estos años de estudio. Realizar, además, una mención especial para Alberto Martín, quien ha sido mi compañero de laboratorio durante la realización del proyecto.

Por último, pero no menos importante, resaltar la inestimable ayuda y paciencia de mis padres, José Manuel y María Bella, así como el apoyo que mi novia, Lara, me ha dado día a día.

Alberto Fernández Añivarro, junio 2018

Resumen

Este trabajo aborda la problemática de la elección, configuración y posterior explotación de una plataforma de gestión de red apropiada para el laboratorio docente del grupo de telemática, que permita identificar fallos en dicha red, reportar datos en forma de gráficas además de generar y notificar alertas personalizadas a los responsables de la gestión del entorno.

Como resultado de una comparación entre diferentes plataformas de gestión de red vigentes en la actualidad, se escoge la plataforma Zabbix, como la solución a implementar en el proyecto. El nodo gestor, basado en Zabbix, se instala en una máquina virtual con sistema operativo Ubuntu, soportada por un sistema de virtualización VMware bajo un sistema anfitrión Windows. Dicho nodo gestor, a través de la interfaz gráfica de Zabbix, permite la monitorización de toda la red del laboratorio. Sobre el resto de ordenadores del laboratorio se instalan agentes nativos de Zabbix, permitiendo así su control y monitorización. La instalación de estos agentes, facilita el acceso a un conjunto de parámetros de los equipos gestionados, como la carga de CPU, el uso de la memoria, etc.

Los equipos de comunicaciones switches o routers, que no permiten la instalación de agentes Zabbix nativos, se comunican con el nodo gestor mediante los agentes SNMP que ya llevan incorporados. Además, en el nodo gestor, se instala un servidor de correo para enviar las notificaciones de errores y fallos a los responsables de gestión. De este modo, el gestor es capaz de recibir alertas sobre lo que está sucediendo en su red, sin necesidad de estar continuamente pendiente de la pantalla de alertas y notificaciones implementada sobre el nodo gestor.

Abstract

This project addresses the problem of the choice, configuration and subsequent exploitation of a network management platform appropriate for the teaching laboratory of the telematics group, to identify faults in the network, report data in the form of graphics as well as generate and notify alerts customized to those responsible for the management of the environment.

As a result of a comparison between different network management platforms currently in force, the Zabbix platform is chosen as the solution to be implemented in the project. The manager node, based on Zabbix, is installed in a virtual machine with an Ubuntu operating system, supported by a VMware virtualization system under a Windows host system. Said managing node, through the Zabbix graphical interface, allows the monitoring of the entire laboratory network. On the rest of the computers of the laboratory, Zabbix native agents are installed, allowing its control and monitoring. The installation of these agents facilitates access to a set of parameters of managed equipment, such as CPU load, memory usage, etc.

The communications equipment switches or routers, which do not allow the installation of native Zabbix agents, communicate with the manager node through the SNMP agents that are already incorporated. In addition, in the manager node, a mail server is installed to send notifications of errors and failures to the managers. In this way, the manager is able to receive alerts about what is happening in their network, without the need to be continually aware of the alerts and notifications screen implemented on the manager node.

ÍNDICE

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

CAPÍTULO 1. PREÁMBULO

Inicialmente, desde el origen del modelo TCP/IP en la década de los setenta se utilizaban herramientas basadas en el protocolo ICMP (Internet-Control Message Protocol) para la realización de la gestión, cuya principal herramienta es el PING (Packet Internet Groper), que permite comprobar la comunicación entre dos máquinas, calcular tiempos medios de respuesta y pérdidas de paquetes. Más adelante, con el crecimiento exponencial de internet surge la necesidad de utilizar herramientas de gestión más potentes. Como resultado, en los años ochenta se recogen varias propuestas de estándares de protocolos de gestión para TCP/IP. De entre esas propuestas resulta elegida SNMP (Simple Network Management Protocol) por ser más simple y necesitar menos esfuerzo para desarrollarse. Su estructura básica consiste en una estación de gestión, agentes de gestión, una base de información de gestión o MIB (Management Information Base) y el protocolo de gestión de red en el que se basa para realizar la comunicación. [1]

La gestión, entendida como tal, es la tarea que cubre todas las precauciones y actividades que aseguren el uso eficiente y efectivo de procesos y recursos distribuidos, los cuales pueden constituir una red de comunicaciones o un sistema distribuido. [2] Por lo tanto, dependiendo de si se da prioridad a la gestión de componentes de una red o a la gestión de un sistema distribuido se diferenciará entre gestión de red (Network Management) y gestión de sistemas (System Management). En la actualidad se tiende hacia un entorno de gestión integrada de redes y sistemas (Integrated Network and Systems Management).

Para llevar a cabo esa tarea de forma eficiente y automatizada, el gestor de la red hace uso de la monitorización. La monitorización de redes es el uso de un sistema que constantemente monitoriza una red de computadoras buscando componentes lentos o fallidos y luego notifica al administrador de esa red (vía correo electrónico, teléfono u otras alarmas) en caso de cortes o fallos. Es un subconjunto de las funciones involucradas en la gestión de redes. [3]

Hoy en día, realizar una monitorización de los diversos componentes de una infraestructura de red se antoja imprescindible tanto para redes del ámbito empresarial como para redes privadas, de investigación o docencia. Esto es debido a la actual necesidad de proporcionar un alto nivel de servicio y disponibilidad, a la vez que el número de dispositivos, tecnologías y plataformas aumentan. Sin embargo, el objetivo de la monitorización no será únicamente el de reaccionar frente a los fallos surgidos en el servicio o la red, sino el de establecer un sistema de control que escale los problemas detectados de manera instantánea. De este modo, los tiempos de parada del funcionamiento de la red se reducen drásticamente, mejorando la calidad de servicio ofrecida.

La solución de monitorización que cumpla con los requisitos especificados debe ser de arquitectura abierta y debe respaldarse en protocolos estándares. De esta manera, podrán monitorizarse plataformas de cualquier tipo y de cualquier fabricante: dispositivos de comunicaciones, estaciones de trabajo, servidores y hosts críticos, impresoras de red, dispositivos de seguridad, PLCs, aplicaciones críticas, etc.

Dado que se cuenta con protocolos estándares (TCP/IP y SNMP) se dependerá de la información que proporcione cada fabricante sobre sus productos. Dicha información reside en ficheros de tipo MIB (Management Information Base) que describen los productos, las alarmas que pueden generar y las consultas sobre rendimiento,

descripción del producto y disponibilidad que pueden realizarse. Por ejemplo, a un switch se le podrá pedir información de cada una de sus conexiones activas.

Es preciso aclarar el hecho de que una plataforma de monitorización no debe sustituir a las herramientas de administración y gestión de los dispositivos. Cada fabricante gestiona sus dispositivos a través de sistemas diferentes. Una misma herramienta difícilmente servirá para generar usuarios, detectar intrusos, analizar protocolos y administrar el correo electrónico. Sin embargo, si resulta tremendamente práctico integrar dichas herramientas a la solución de monitorización de manera que cada dispositivo responda a un cierto tipo de acciones y conduzca así a las herramientas y utilidades necesarias para gestionarlo o solucionar los problemas detectados en el mismo. [4]

Para escoger adecuadamente la plataforma que mejor se adapte a la red a gestionar habrá que tener en cuenta características como su coste, la facilidad o complejidad de su instalación, el uso de "plugins" que completen la herramienta, la posibilidad de manejar una interfaz web, el uso de plantillas y la existencia de foros con muchos miembros activos que garantice la resolución de las dudas que puedan surgir con su utilización.

El entorno sobre el que trabaja la plataforma de monitorización que se propone en este trabajo es el de una red orientada a la docencia. Por ello, se debe tener en cuenta el hecho de que la propia red va a ser utilizada por multitud de estudiantes durante sus prácticas para fines totalmente diferentes. La monitorización de esta red permite garantizar el buen funcionamiento de los equipos que la componen y adelantarse a los posibles fallos que surjan durante su uso casi continuado en el transcurso del período lectivo.

1.2 Motivación y objetivos

La motivación para llevar a cabo este proyecto surge de la necesidad de gestionar la red privada del laboratorio docente de telemática de la escuela de ingenieros industriales y de telecomunicación de la Universidad de Cantabria, optimizando a su vez el rendimiento de cada uno de sus componentes y previniendo contra fallos o problemas causados por servidores caídos, conexiones de red, cuellos de botella, etc. Se trata de un laboratorio que tiene un uso continuado a lo largo del período lectivo y que cuenta con diferentes usuarios que, por lo tanto, le dan un diferente uso. Por todo ello es interesante realizar un seguimiento y una gestión del mismo para garantizar que todo funciona correctamente y conseguir anticiparse así a fallos graves que obliguen a la parada en su totalidad de los equipos.

El objetivo principal de este proyecto es la elección, instalación y finalmente uso de una plataforma de monitorización que permita llevar a cabo las tareas de gestión necesarias para el buen funcionamiento de este laboratorio.

La elección de la plataforma de monitorización se realiza a través de una comparativa de diferentes herramientas teniendo en cuenta el entorno particular del laboratorio, sus necesidades y particularidades.

La instalación y el uso de la plataforma elegida constituye la mayor parte del documento y, por lo tanto, el objetivo indispensable a lograr en este proyecto.

A su vez se diferencian una serie de objetivos secundarios que se corresponden con las distintas funcionalidades que se desean implementar en el entorno de monitorización como son el familiarizarse con la instalación, manejo y configuración de la herramienta

de monitorización, comprobar la viabilidad de la instalación de la herramienta sobre la infraestructura del laboratorio de telemática, configurar la plataforma para el envío de alertas, realizar gráficas personalizadas que reflejen los datos que se quieren conocer de los equipos, realizar mapas de red que den una visión global de la topología de la red que se monitoriza, reconocer y solventar los problemas surgidos durante la implementación y analizar los resultados obtenidos en la monitorización de los eventos.

1.2 Estructura del documento

El proyecto está compuesto por cinco capítulos.

Capítulo 1: Preámbulo. En este capítulo se hace una breve introducción y se exponen los objetivos a cumplir, además de darse una visión global de la estructura del documento.

Capítulo 2: Aspectos teóricos. Se da una pequeña introducción de la gestión de redes, exponiéndose además la plataforma de monitorización escogida, en este caso Zabbix, detallando su funcionamiento y características.

Capítulo 3: Aspectos prácticos. Este capítulo se basa en la preparación previa del entorno de trabajo, desde la obtención de las máquinas virtuales en las que se trabaja hasta la instalación de los requisitos previos y la obtención de la plataforma elegida.

Capítulo 4: Implementación en el laboratorio. Se aborda la configuración de la plataforma de monitorización junto con los equipos físicos que componen el laboratorio para que el intercambio de datos entre ambos sea posible.

Capítulo 5: Conclusiones y líneas futuras. Por último, se valora el resultado final del proyecto y se realizan observaciones acerca de los problemas e inconvenientes surgidos, así como las posibles mejoras a implementar en proyectos futuros.

CAPÍTULO 2. ASPECTOS TEÓRICOS

2.1 La gestión de redes

La gestión de una red cualquiera consiste en la monitorización y el control de los recursos de la propia red con el fin de evitar un mal funcionamiento, y, por lo tanto, una degradación de sus prestaciones. Hoy en día la necesidad de llevar a cabo esta gestión se antoja imprescindible ya que la información manejada tiende a ser mayor y estar más dispersa. Además, en la mayoría de redes se encuentran productos y servicios de múltiples fabricantes (redes heterogéneas), lo que hace que centralizar la gestión en un único sistema aporte comodidad a la ejecución de la tarea. La finalidad de la gestión es asegurar un servicio casi continuo a los usuarios finales de la red.

Un sistema de gestión está compuesto por un gestor, que emite las directivas de operaciones de gestión y recibe notificaciones y respuestas, un agente, que tiene la función de responder a las directivas enviadas por el gestor, el protocolo mediante el cual se lleva a cabo la comunicación, que es el conjunto de especificaciones y convenciones que gobiernan la interacción de los procesos y los elementos dentro del sistema y la base de información de gestión o MIB (Management Information Base), que almacena los datos de los dispositivos o componentes de la red.

El protocolo más común a utilizar en este tipo de redes es SNMP (Protocolo de gestión simple de red), que pertenece al conjunto de protocolos TCP/IP, ya que la gran mayoría de los equipos de telecomunicación lo soportan. [5] Es un protocolo que les permite a los administradores de la red administrar dispositivos de red y diagnosticar sus problemas.

En general, un sistema de gestión de red se basa en dos elementos principales: un supervisor y un conjunto de agentes. El supervisor es el terminal que le permite al gestor de red realizar solicitudes de gestión. Los agentes son entidades que se encuentran al nivel de cada interfaz. Ellos conectan a la red los dispositivos gestionados y permiten recopilar información sobre los diferentes objetos a gestionar.

Los conmutadores, concentradores (“hubs”), “routers” y servidores son ejemplos de hardware que contienen objetos de gestión. Estos objetos, pueden ser información de hardware, parámetros de configuración, estadísticas de rendimiento y demás elementos que estén directamente relacionados con el comportamiento en progreso del hardware en cuestión. Todos ellos se encuentran clasificados en algo similar a una base de datos denominada MIB (Base de datos de información de gestión). El protocolo SNMP permite el diálogo entre el supervisor y los agentes para recolectar los objetos requeridos en la MIB. [6]

El protocolo SNMP tiene dos formas de funcionar: “polling” y “traps”. El “polling” consiste en lanzar consultas remotas de forma activa o a demanda, realizando una operación síncrona de consulta. Los “traps” son mensajes que envían los dispositivos SNMP gestionados, de forma asíncrona al nodo gestor basándose en cambios o eventos. Al configurar un sistema de monitorización SNMP utilizamos ambos modos de trabajo del protocolo. [7]

En definitiva, una red SNMP está compuesta por tres elementos principales. Los recursos gestionados, los agentes y los sistemas de administración de red (NMS), que crean una conexión y ejecutan aplicaciones de monitorización y control de los elementos conectados.

En cuanto a la MIB, como ya se ha comentado, se trata de un tipo de base de datos que contiene información jerárquica, estructurada en forma de árbol, de todos los parámetros gestionables en cada dispositivo gestionado de una red de comunicaciones. Define las variables usadas por el protocolo SNMP para supervisar y controlar los componentes de una red. Está compuesta por una serie de objetos que representan los dispositivos (como enrutadores y conmutadores) en la red. Cada objeto manejado en una MIB tiene un identificador de objeto único (OID) e incluye el tipo de objeto (un contador, un medidor o "gauge", etc.), el nivel de acceso (solo lectura o lectura/escritura), restricciones de tamaño, y la información del rango de valores que podrá tomar el objeto. [8]

Figura 1. Esquema de un sistema de gestión

En la Figura 1 se puede ver cómo se establecen las relaciones entre los distintos elementos del sistema de gestión de forma esquematizada donde tanto el gestor como el agente poseen una MIB asociada y se especifica la comunicación entre ambos mediante el protocolo de gestión.

La base del funcionamiento de los sistemas de apoyo a la gestión reside en el intercambio de información entre los nodos gestores y los nodos gestionados. Es lo que se denomina paradigma gestor-agente. Siendo los nodos gestores aquellos nodos de la red que poseen un gestor y los gestionados, aquellos que poseen un agente.

El gestor pide al agente, a través de un protocolo de gestión de red (SNMP), que realice determinadas operaciones con los datos de gestión, gracias a las cuales podrá conocer el estado del recurso e influir así en su comportamiento.

En la Figura 2 se realiza una descripción de los diferentes sub-modelos que forman la arquitectura de gestión de la red. Esta arquitectura se organiza en cuatro sub-modelos: el **modelo de comunicación**, que establece el funcionamiento del intercambio de información entre los diferentes componentes del sistema de gestión; el **modelo de información**, que obtiene los datos e información que facilitan el funcionamiento y uso de la red de comunicaciones; el **modelo de organización**, que establece los diferentes papeles o funciones dentro del sistema de gestión, así como su distribución espacial y en el que se ha de hablar de la filosofía gestor-agente como la más común. El principio de funcionamiento de esta forma de organización reside en el intercambio de información de gestión entre nodos gestores y nodos gestionados. Habitualmente, los agentes mantienen en cada nodo gestionado información acerca del estado y las características de funcionamiento de un determinado recurso de la red. El gestor pide al agente, a través de un protocolo de gestión de red, que realice determinadas operaciones con estos datos de gestión, gracias a las cuales podrá conocer el estado del recurso y podrá influir en su comportamiento; y el **modelo funcional** que se encarga de definir las funciones específicas de un sistema de gestión, las cuales se pueden dividir en cinco áreas funcionales, la gestión de configuración, de rendimiento, de contabilidad, fallos y seguridad. [9]

- El objetivo de la gestión de configuración es obtener datos de la red y utilizarlos para incorporar, mantener o retirar los distintos componentes y recursos.
- La gestión de prestaciones o del rendimiento tiene como objetivo principal el mantenimiento del nivel de servicio que la red ofrece a sus usuarios.
- La gestión de contabilidad tiene como misión la medida de parámetros de utilización de la red que permitan a su explotador preparar las correspondientes facturas a sus clientes.
- La gestión de fallos tiene por objetivo fundamental la localización y recuperación de los problemas de la red.
- La gestión de la seguridad debe ofrecer mecanismos que faciliten el mantenimiento de políticas de seguridad orientadas a la protección contra ataques de intrusos. [9]

El encargado de gestionar la red debe escoger la plataforma de gestión a utilizar que resuelva la problemática de gestión en cada una de las áreas anteriormente comentadas.

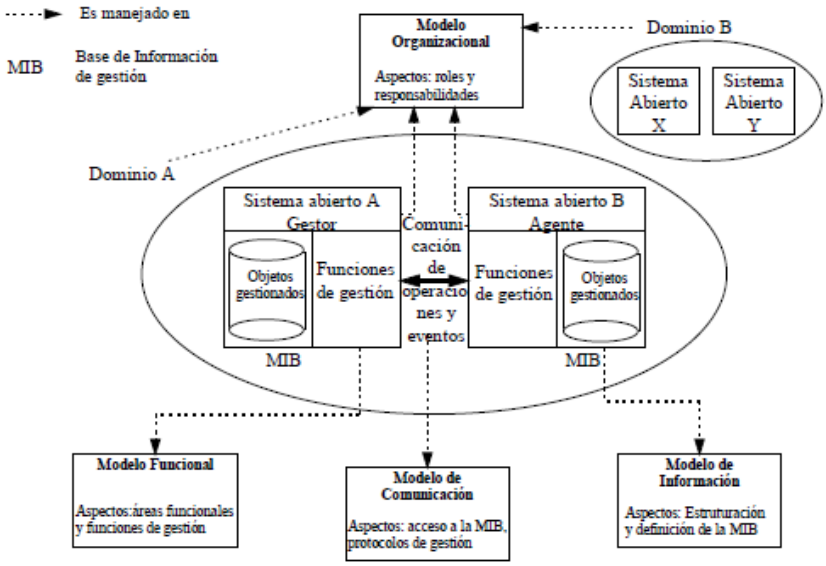


Figura 2. Esquema de los sub-modelos de la arquitectura de gestión

2.2 Gestión con o sin agentes

La persona encargada de la gestión de la red puede elegir entre dos formas de obtener los datos de los equipos de su infraestructura dependiendo de sus necesidades y objetivos. Puede hacerlo mediante una monitorización basada en agentes, en la que es necesario realizar la instalación de un pequeño programa o aplicación en la propia máquina, o mediante una monitorización sin agentes.

El término sin agentes o “agentless”, en el contexto de la gestión de redes, se refiere a las operaciones en las que no se necesita ejecutar ningún servicio, daemon o proceso en segundo plano en la máquina. La supervisión sin agente se implementa de una de estas dos formas:

Usando una API remota expuesta por la plataforma o servicio que se monitoriza o analizando directamente los paquetes de red que fluyen entre los componentes del servicio.

En la monitorización sin agentes, no hay necesidad de instalar o ejecutar un nuevo software relacionado con la tarea en sí y los programas sin agentes acceden directamente a los archivos.

A la hora de hablar de sus características más positivas hay que resaltar una implementación más simple (no necesariamente menos costosa), debido a la ausencia de un agente y un menor gasto, en la mayoría de los casos, en los recursos de la máquina administrada.

Por el contrario, hay que decir que esta forma de gestión supone un aumento en la carga de la red debido a un sondeo constante. Conlleva, además, la aparición de falsos negativos, por lo que una interrupción de la red tiene el mismo resultado cuando la máquina objetivo está inactiva.

Por otro lado, la supervisión basada en agentes normalmente implica la instalación de un agente en, o junto al sistema que queremos monitorizar. Este agente a menudo es suministrado por un proveedor con una solución de monitoreo. Un sistema de supervisión basado en agentes recopila automáticamente métricas sobre el rendimiento y la disponibilidad de recursos de hardware, sistemas operativos y aplicaciones en entornos físicos, virtuales y en la nube. Por lo tanto, podemos decir que la supervisión basada en agentes se lleva a cabo donde el objetivo principal es la supervisión y la gestión en profundidad. Entre sus ventajas destacan:

El hecho de que se pueden descartar datos poco interesantes, lo que conlleva una carga de red reducida.

El agente local continúa actuando incluso cuando la red no funciona, por lo que la administración del sistema no se suspende por una falla de la red. Se eliminan los falsos negativos.

Los agentes pueden estar estrechamente integrados, es decir, un agente local puede realizar más fácilmente una acción correctiva automática.

Poca o ninguna dependencia, dado que la administración es local, el estado de otros dispositivos tiene poco o ningún efecto en la máquina objetivo.

Entre sus principales desventajas, se ha de mencionar el hecho de que el software debe instalarse en todas las máquinas, lo que conlleva un mayor despliegue y posibles efectos de interacción negativa con otro software en la máquina. Además, un agente local inteligente utiliza recursos del sistema. El agente local es una aplicación en la máquina administrada, por lo que, como cualquier otra aplicación, consumirá recursos. Estos agentes se tendrán que actualizar a medida que se actualice la plataforma de monitorización, por lo que en grandes infraestructuras esto conlleva una tarea bastante costosa.

Unos buenos ejemplos para la supervisión sin agentes son Windows Management Instrumentation (WMI) y Simple Network Management Protocol (SNMP). WMI se usa para monitorizar y administrar Microsoft Windows, mientras que SNMP generalmente se usa de manera más genérica para monitorizar y administrar sistemas Linux y Unix, entornos de red y otros dispositivos.

En términos de implementación: la monitorización sin agentes es más fácil de implementar que la basada en agentes, ya que en esta los agentes deben implementarse en cada servidor.

Sin embargo, la supervisión sin agente requiere tráfico de red adicional a medida que los datos de rendimiento sin procesar se transportan a un recopilador de datos remoto. Mientras que la supervisión basada en agentes es eficiente en el ancho de banda porque los datos se recopilan localmente.

En conclusión, se puede decir que tanto la supervisión basada en agentes como la sin agentes pueden satisfacer las necesidades de diferentes usuarios. Es por ello recomendable que se analicen los requisitos de supervisión, para acceder así a las opciones adecuadas para las necesidades de la red. Algunos dispositivos no permiten la instalación de un agente; por ejemplo, los dispositivos de red y almacenamiento como "switches" o "routers", por lo que en esos casos no habrá lugar a elección. [19]

2.3 Plataformas de gestión

Históricamente, la gestión de red se realiza mediante un conjunto de programas aislados, cada uno encargado de gestionar un conjunto específico de componentes (dispositivos o datos de gestión) de la red, pero las restricciones de coste, espacio físico y disponibilidad de técnicos plantean la necesidad de una gestión integrada desde un solo sistema, que debería presentar sus interconexiones en un mapa de la red.

Una plataforma de gestión de red es una aplicación software que proporciona la funcionalidad básica de gestión de red para los diferentes componentes de una red. El objetivo de la plataforma es proporcionar una funcionalidad genérica para gestionar dispositivos de red diversos. [10]

Actualmente existe una gran variedad y competencia en cuanto a las herramientas de apoyo para la gestión de redes se refiere. Por esta razón se reduce la exposición a tres debido a su posible idoneidad con respecto al entorno de trabajo que se utiliza en este proyecto.

2.3.1 Nagios

Nagios es un sistema de monitorización de redes de código abierto, publicado bajo la licencia GPL (General public license), que vigila los equipos (hardware) y servicios (software) que se especifiquen, alertando cuando el comportamiento de estos no sea el deseado. [11] Además, acepta los informes de estado de otros procesos o equipos, por ejemplo, un servidor web puede informar directamente a Nagios si no está sobrecargado.

La monitorización de sistemas en Nagios se divide en dos categorías de objetos: hosts o equipos y servicios. Los equipos representan un dispositivo físico o virtual en la red (servidores, routers, estaciones de trabajo, impresoras, etc.) Los servicios son funcionalidades específicas, por ejemplo, un servidor SSH (Secure Shell) puede definirse como un servicio monitorizado. Esta herramienta solamente utiliza cuatro diferentes estados: "Ok", "Warning", "Critical", y "Unknown". Este sistema de comprobaciones está basado en "plugins", lo que significa que, si se quiere monitorizar algo que todavía no es posible, se puede escribir un pequeño código que lo consiga. [12]

Entre sus características principales destacan la monitorización integral, es decir, que permite monitorizar aplicaciones, servicios, sistemas operativos, protocolos de red, métricas del sistema y componentes de infraestructura con una sola herramienta. Cuenta además con reconocimientos de alerta que proporcionan comunicación sobre problemas conocidos y respuestas a problemas, informes históricos que proporcionan un registro de alertas, notificaciones, interrupciones y respuesta de alerta. Posee una

gran comunidad de usuarios, una arquitectura extensible y la capacidad de ofrecer una vista centralizada de toda la infraestructura de TI (tecnologías de la información) monitorizada. Está publicado bajo la licencia GPL (General public license). [13]

2.3.2 Pandora

Pandora FMS es un software de código abierto que sirve para monitorizar y medir todo tipo de elementos de red. Monitoriza sistemas, aplicaciones o dispositivos de red. Permite conocer el estado de cada elemento de un sistema a lo largo del tiempo ya que dispone de histórico de datos y eventos. Pandora FMS está orientado a grandes entornos, y permite gestionar con y sin agentes, varios miles de sistemas, por lo que se puede emplear en grandes “clusters”, centros de datos y redes de todo tipo. Puede, además, recoger información de cualquier sistema operativo, con agentes, específicos para cada plataforma, que recolectan datos y los envían al servidor. [14]

Sus características más destacadas son el descubrimiento de los elementos que forman parte de la red de manera sencilla, la reducción de costes aprovechando al máximo el CPD (Centro de Procesamiento de Datos), detección de cuellos de botella y rediseño de la red acorde a las necesidades de los sistemas, la detección de la aparición de nuevos componentes automáticamente junto con su localización y ataques de seguridad. [15]

2.3.3 Zabbix

Zabbix es un software diseñado para supervisar la disponibilidad y el rendimiento de los componentes de una infraestructura TI (Tecnología de la Información). Con él, se es capaz de realizar un monitoreo en tiempo real de servidores, máquinas virtuales y dispositivos de red simultáneamente. Además, junto con el almacenamiento de los datos, las funciones de visualización están disponibles (vistas generales, mapas, gráficos, pantallas, etc.), así como formas muy flexibles de analizar los datos con el fin de alertar. Ofrece un gran rendimiento para la recopilación de datos y se puede escalar a entornos muy grandes, aunque en el caso de este proyecto el entorno es de un tamaño reducido.

Zabbix viene con una interfaz basada en web, autenticación de usuario segura y un esquema flexible de permisos de usuario, tiene agentes nativos que recopilan datos de prácticamente cualquier sistema operativo, aunque los métodos de monitoreo sin agentes también están disponibles.

A todo esto, se puede añadir que, Zabbix, puede detectar automáticamente los servidores y dispositivos de red, así como un descubrimiento “low-level” o de bajo nivel con métodos para asignar automáticamente comprobaciones de rendimiento y disponibilidad a las entidades descubiertas. [16]

Sus principales características son la capacidad de monitorizar todo aquello que se encuentra dentro de su red, como rendimiento y disponibilidad de servidores, aplicaciones WEB, bases de datos, equipos de red, etc. Está diseñado tanto para entornos grandes como pequeños y tiene una disponibilidad total (24/7). Cuenta con monitoreo proactivo, el cual reduce costos operativos al evitar el tiempo de inactividad, planificación de la capacidad para mantener la eficiencia a medida que el entorno crece o disminuye. Posee código abierto en su totalidad y está publicado bajo la licencia GPL. [17]

2.3.4 Comparativa y elección de la plataforma

La principal diferencia a tener en cuenta entre estas tres herramientas es el hecho de que mientras Zabbix y Pandora FMS poseen una gestión de la interfaz basada en web, Nagios emplea “scripts” y demás procedimientos manuales que requieren la utilización de herramientas de terceros para su implementación. Sin embargo, la comunidad con mayor tamaño se encuentra en Nagios ya que lleva mucho más tiempo en funcionamiento que las otras dos herramientas. Esto hace que sea mucho más fácil encontrar información en internet, foros, etc.

A continuación, se realiza una comparación más detallada, que incluye costo, configuración, instalación, administración y muchos otros requisitos importantes que se necesitan en un paquete de software de monitoreo de red.

En primer lugar, se ha de resaltar que tanto Zabbix como Nagios y Pandora (estos dos últimos en su versión estándar) no solo son de código abierto, también son gratuitos, ya que el código abierto no siempre significa gratis. Sin embargo, hay ligeras diferencias entre estas tres soluciones. Zabbix es completamente gratuito, de código abierto y no tiene una edición Enterprise por separado como ocurre en el caso de Nagios y Pandora.

Uno de los principales desafíos que los nuevos usuarios tienen con Nagios es el hecho de que tienen que hacer toda la configuración en archivos de texto, desde el archivo de configuración principal hasta la configuración requerida para definir los hosts y servicios a ser monitorizados. Esto no ocurre así en Zabbix y Pandora los que se administran principalmente a través de su interfaz web, lo que significa que la curva de aprendizaje no es tan pronunciada.

Aunque Nagios está configurado con archivos de texto, también cuenta con una interfaz web. Sin embargo, la interfaz web de Nagios es muy rudimentaria, parece obsoleta y es básicamente de solo lectura. Además, todo lo que se puede hacer con ella es ver los hosts, los servicios monitorizados y generar informes, por lo que el término que mejor describe la interfaz es el de “solo visualización”, ya que no permite configurar nada.

La interfaz web de Zabbix tiene pestañas y sub-pestañas ordenadas que le permiten navegar por las diferentes partes de la aplicación, y, aparte de esto, configurarla.

Una de las mejores características que tienen Zabbix y Pandora es el hecho de incluir plantillas que se pueden usar para monitorizar diferentes servicios. Esto hace que la configuración sea más rápida y fácil. Nagios, por otro lado, no proporciona esas plantillas. Esto puede no ser necesariamente un problema debido a la gran comunidad de usuarios que Nagios posee, y por lo que es probable que se encuentre lo que se necesita con una búsqueda rápida en la red. Para suplir muchas de esas carencias, Nagios hace uso de “plugins” o complementos, que son aplicaciones informáticas que añaden funcionalidades adicionales al software original. Algunos de estos complementos son necesarios en Nagios para implementar funcionalidades que Zabbix ofrece nativamente (por ejemplo, gráficos). Sin embargo, el hecho de que Nagios posea tantos complementos hace de él una plataforma de monitorización altamente personalizable.

Una diferencia distintiva entre Nagios y, Zabbix y Pandora es la disponibilidad de gráficos en Zabbix y Pandora, pero la falta de ellos en Nagios. Como se puede imaginar, los gráficos son una forma muy útil de ver datos porque pueden mostrar información histórica de una manera fácil y amigable para el usuario.

En conclusión, a la hora de elegir entre estas tres plataformas, Zabbix resulta la escogida debido a su fácil implementación, el hecho de que se pueda administrar vía interfaz web y el poseer gran cantidad de plantillas y gráficos. [18]

Todas estas y más características se recogen en la Tabla 1 de forma detallada para dar una visión global de las tres plataformas.

Nombre	**Nagios**	**Zabbix**	**Pandora FMS**
Informes IP SLAs	Vía Plugin	Si	Si
Agrupación lógica	Si	Si	Si
Tendencias	Si	Si	Si
Predicción de tendencias	No	Si	Si
Autodescubrimiento	Vía Plugin	Si	Si
Sin agentes	Soportado	Soportado	Soportado
SNMP	Vía Plugin	Si	Si
Syslog	Vía Plugin	Si	Si
Plugins	Si	Si	Si
Triggers	Si	Si	Si
WebApp	Si	Control Total	Control Total
Monitoreo distribuido	Si	Si	Si
Inventario	Vía Plugin	Si	Si
Plataforma	C	C PHP	Perl PHP C++ Java
Método de almacenaje de datos	FlatFile SQL MySQL	Oracle MySQL IBM DB2 SQLite	MySQL Oracle
Licencia	GPL	GPL	GPLv2 Comercial
Mapas	Si	Si	Si
Control de acceso	Si	Si	Si

IPv6	Si	Si	Si

Tabla 1. Comparativa de plataformas

Leyenda:

- Informe IP SLAs: Soporte del mecanismo del acuerdo de nivel de servicio IP de CISCO
- Agrupación lógica: Soporta organizar los hosts o dispositivos que monitorea en grupos definidos por el usuario
- Tendencias: Proporciona tendencias de datos de red a lo largo del tiempo
- Predicción de tendencias: El software presenta algoritmos diseñados para predecir las futuras estadísticas de la red
- Autodescubrimiento: El software descubre automáticamente los hosts o dispositivos de red a los que está conectado
- Sin agentes: El producto no se basa en un agente de software que se debe ejecutar en los hosts que está supervisando, por lo que los datos se pueden enviar a un servidor central. “Soportado” significa que se puede usar un agente, pero no es obligatorio. Un “daemon SNMP” no cuenta como agente
- SNMP: Capaz de recuperar e informar sobre las estadísticas de SNMP
- Syslog: Capaz de recibir e informar sobre “syslogs”
- Plugins: Arquitectura del software basada en una serie de “complementos” que proporcionan funcionalidad adicional a la soportada de forma nativa
- Triggers: Capaz de detectar valores por encima de un umbral en datos de red y alertar al administrador de alguna forma
- WebApp: “Full Control”: Todos los aspectos del producto pueden ser controlados a través de “web-based frontend” incluyendo tareas de mantenimiento de bajo nivel como configuración de software y actualizaciones
- Monitoreo distribuido: Capaz de aprovechar más de un servidor para distribuir la carga de la red de monitoreo
- Inventario: Mantiene un registro de inventario de hardware y/o software para los hosts y dispositivos que monitorea
- Plataforma: La plataforma (lenguaje de codificación) en la que se desarrolló/escribió la herramienta
- Método de almacenaje de datos: Método principal utilizado para almacenar los datos de la red que monitorea
- Mapas: Presenta mapas gráficos en red que representan los hosts y dispositivos que monitorea y los vínculos entre ellos

- Control de acceso: Ofrece seguridad a nivel de usuario, permitiendo a un administrador impedir el acceso a ciertas partes del producto por usuario o por función

- IPv6: Soporta monitorización de hosts y/o dispositivos IPv6, recepción de datos IPv6 y ejecución en un servidor habilitado para IPv6. Soporta comunicación mediante IPv6 con el agente SNMP a través de una dirección IPv6.

2.4 Plataforma Zabbix

Zabbix es un Sistema de Monitorización de Redes creado por Alexei Vladishev, y actualmente, la empresa Zabbix SIA, lo desarrolla y respalda activamente. Se trata de un software que monitoriza numerosos parámetros de una red, así como el estado e integridad de los servidores. Zabbix utiliza un mecanismo de notificación flexible que permite a los usuarios configurar alertas basadas en correo electrónico para prácticamente cualquier evento. Esto permite una reacción rápida a los problemas del servidor. Zabbix ofrece excelentes funciones de informes y visualización de datos basadas en los datos almacenados. Esto hace que Zabbix sea ideal para la planificación de la capacidad de la red.

Zabbix admite, además, tanto sondeo como captura. Todos los informes y estadísticas que son generados por Zabbix, así como los parámetros de configuración, se acceden a través de una interfaz basada en la web, la cual garantiza que el estado de su red y el estado de sus servidores se puedan evaluar desde cualquier ubicación. Con una configuración adecuada, Zabbix puede desempeñar un papel importante en la monitorización de la infraestructura de TI (Tecnologías de la Información). Esto es igualmente cierto tanto para organizaciones pequeñas con algunos servidores como para grandes empresas con una multitud de servidores.

Con respecto a su contexto histórico, cabe resaltar que Zabbix fue iniciado como un proyecto interno de software en 1998. Después de 3 años, en 2001, este fue lanzado al público sobre la licencia pública general de GNU o más conocida por su nombre en inglés GNU General Public License, que es una licencia de derecho de autor muy usada en el mundo del software libre y código abierto y garantiza a los usuarios finales la libertad de usar, estudiar, compartir y modificar el software. Actualmente cumple los términos de la versión 2 de la GNU GPL. Pasaron 3 años más hasta su primera versión estable, 1.0, que fue lanzada en 2004. Este proyecto se desarrolla bajo la última versión más estable, la versión 3.4.

Zabbix usa MySQL, PostgreSQL, SQLite, Oracle o IBM DB2 como base de datos. Su backend está escrito en C y el frontend web en PHP.

Para ser capaz de manejarse con el entorno de Zabbix es imprescindible conocer algunos conceptos básicos de la plataforma, como:

- Host: Un dispositivo en red que se desea monitorizar.
- Host group: Una agrupación lógica de hosts; puede contener hosts y plantillas. Los hosts y las plantillas dentro de un grupo de host no están de ninguna manera vinculados entre sí. Los grupos de host se utilizan cuando se asignan derechos de acceso a hosts para diferentes grupos de usuarios.
- Item: Un dato particular que se desea recibir de un host, una métrica de datos.
- Trigger: Una expresión lógica que define un umbral de problema y se utiliza para "evaluar" los datos recibidos en los elementos. Cuando los datos recibidos están por encima del umbral, los desencadenantes pasan de 'Ok' a 'Problema'. Cuando

los datos recibidos están por debajo del umbral, los activadores permanecen en o regresan a un estado 'Ok'.

- Event: La ocurrencia de algo que merece atención, como un estado de cambio de un trigger o un registro automático de descubrimiento.
- Problem: Un trigger que se encuentra en estado de “Problema”.
- Action: Un medio predefinido de reaccionar a un evento. Una acción consiste en operaciones (por ejemplo, enviar una notificación) y condiciones (cuando se lleva a cabo la operación).
- Escalation: Un escenario personalizado para ejecutar operaciones dentro de una acción; una secuencia de envío de notificaciones y ejecución de comandos remotos.
- Media: Un medio para entregar notificaciones; canal de entrega.
- Notification: Un mensaje sobre algún evento enviado a un usuario a través del canal de medios elegido.
- Remote command: Un comando predefinido que se ejecuta automáticamente en un host monitoreado con alguna condición.
- Template: Un conjunto de entidades (elementos, disparadores, gráficos, pantallas, aplicaciones, reglas de descubrimiento de bajo nivel, escenarios web) listas para ser aplicadas a uno o varios hosts. El trabajo de las plantillas es acelerar el despliegue de tareas de monitoreo en un host; también para facilitar la aplicación de cambios masivos a las tareas de supervisión. Las plantillas están vinculadas directamente a hosts individuales.
- Application: Una agrupación de items en un grupo lógico.
- Web scenario: Una o varias solicitudes HTTP para verificar la disponibilidad de un sitio web.
- Frontend: La interfaz web provista con Zabbix.
- Zabbix API: Permite usar el protocolo JSON RPC para crear, actualizar y buscar objetos de Zabbix (como hosts, elementos, gráficos y otros) o realizar cualquier otra tarea personalizada.
- Zabbix server: Un proceso central de software Zabbix que realiza monitoreo, interactúa con representantes y agentes de Zabbix, calcula desencadenantes y envía notificaciones; un depósito central de datos.
- Zabbix agent: Un proceso implementado en objetivos de monitoreo para monitorear activamente recursos y aplicaciones locales.
- Zabbix proxy: Un proceso que puede recopilar datos en nombre del servidor de Zabbix, quitando algo de carga de procesamiento del servidor.

La plataforma consta de varios componentes principales de software, cuyas responsabilidades se detallan a continuación.

El servidor Zabbix es el componente central al que los agentes informan y reportan las estadísticas de disponibilidad e integridad. El servidor es, además, el repositorio central en el que se almacenan todos los datos de configuración, estadísticos y operativos.

Para un acceso fácil a Zabbix desde cualquier lugar y desde cualquier plataforma, este proporciona la interfaz basada en la web. La interfaz es parte del servidor de Zabbix, y generalmente (pero no necesariamente) se ejecuta en la misma máquina física que la que ejecuta el servidor.

El proxy de Zabbix puede recopilar datos de rendimiento y disponibilidad en nombre del servidor de Zabbix. Un proxy es una parte opcional de la implementación de Zabbix; sin embargo, puede ser muy beneficioso distribuir la carga de un único servidor Zabbix.

Los agentes Zabbix se implementan en los objetivos de monitorización para monitorizar activamente los recursos locales y las aplicaciones y reportar así los datos recopilados al servidor Zabbix. Una vez estos agentes son instalados se pueden agregar los equipos y configurar sus “ítems”, “triggers”, gráficos, mapas, plantillas, envío de notificaciones, “screens” y “dashboards”.

2.4.1 Creación de hosts

Para agregar un equipo o “host” se debe seleccionar dentro del “frontend” de Zabbix en la pestaña “Configuración” del menú principal y, a su vez, en la opción “Equipos” del menú secundario. Una vez hecho esto, se puede añadir un nuevo equipo pinchando en el botón “Crear equipo” en la parte derecha de la pantalla, así como se puede ver en la Figura 3.

Figura 3. Creación de un host: Paso 1

A continuación, se rellenan los campos con los datos correspondientes al elemento que se quiere crear. Como muestra la Figura 4, estos campos constan de un nombre para el equipo que se quiere agregar, un nombre “visible” que será el nombre por el que se reconozca al equipo dentro de la interfaz, un grupo (es imprescindible que todos los equipos pertenezcan a un grupo) y su dirección IP. En el caso particular de la Figura 4 se trata de un ejemplo para un switch controlado con el protocolo SNMP. Pero, para cualquier otro equipo controlado por un agente Zabbix la configuración será similar con la particularidad de introducir su dirección IP en el campo “Interfaces del agente” y no en “SNMO interfaces”.

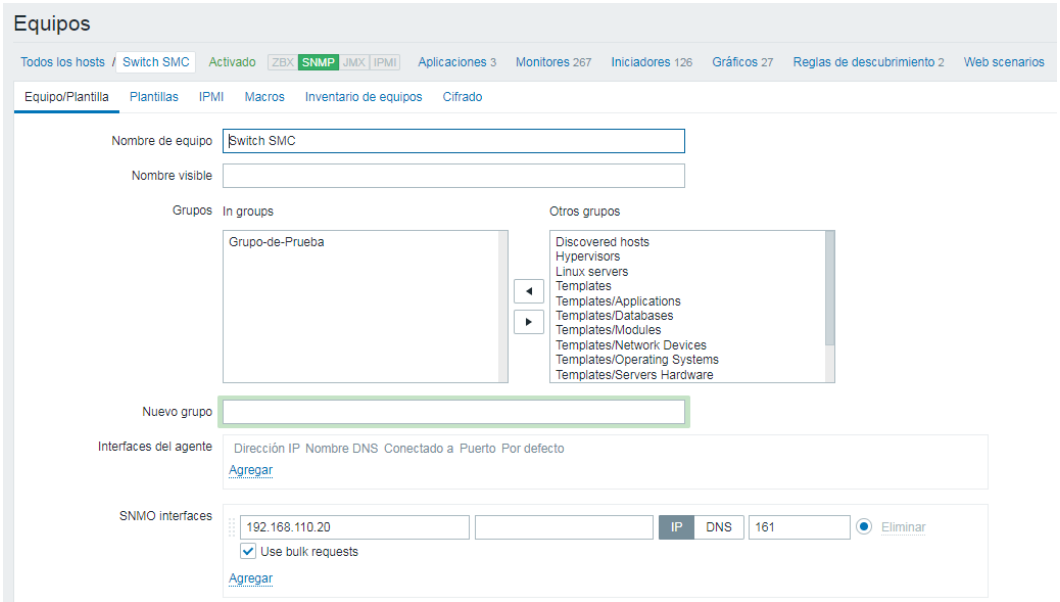


Figura 4. Creación de un host: Paso 2

Además, a la hora de realizar una rápida configuración de este nuevo equipo creado se le puede añadir una determinada plantilla que incluya consigo gráficos, “ítems” y “triggers”. Esto se hace seleccionando la pestaña “Plantillas” del menú secundario como muestra la Figura 5.

En el campo “Link new templates” se selecciona la plantilla que mejor se corresponda con las características del equipo, de entre todas las que ofrece la herramienta Zabbix por defecto.

Una vez añadido el nuevo equipo se procede a comprobar que su estado de monitorización es activo (color verde). Para ello, se debe esperar unos breves instantes para que Zabbix lo reconozca y volver a la pestaña “Configuración” “Equipos” fijándose en la parte derecha de la pantalla donde se verá si el funcionamiento es el correcto. Dicho procedimiento se muestra en la Figura 6.

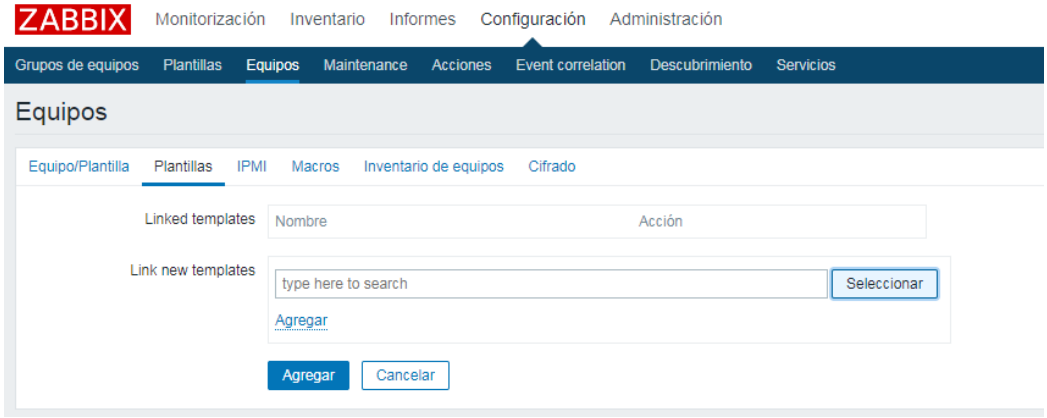


Figura 5. Creación de un host: Paso 3

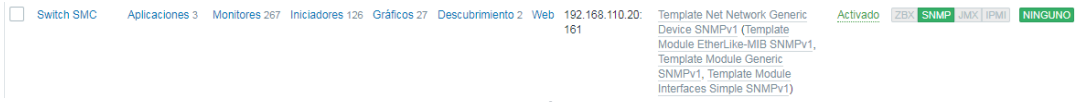


Figura 6. Creación de un host: Paso 4

Por el contrario, cuando el funcionamiento no es el correcto, o simplemente el equipo se encuentra apagado o es inalcanzable por Zabbix, el color que se muestra es el rojo, como se puede observar en la Figura 7.

Figura 7. Creación de un host: Paso 5

2.4.2 Creación de “ítems”

Los “ítems” son los encargados de recolectar la información de los equipos. Para su creación, se debe de ingresar en la pestaña principal “Configuración” y la secundaria “Equipos”. Una vez allí se hace clic en la opción “Monitores” (ya que es la traducción al español que Zabbix hace de ítem) del equipo en el que se quiere crear el “ítem”, esto se muestra en la Figura 8.

Figura 8. Creación de un ítem: Paso 1

Una vez dentro de la pantalla “Monitores” se procede a crear uno haciendo clic en “Crear monitor”, parte derecha de la pantalla como muestra la Figura 9.

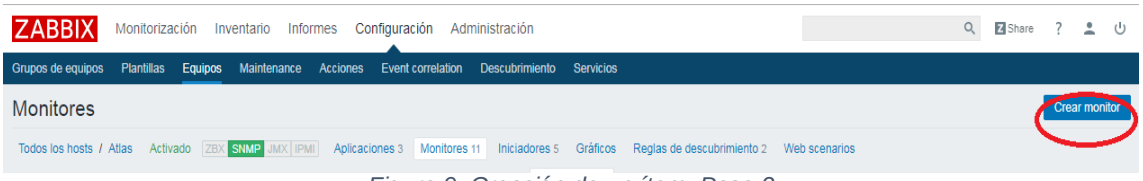


Figura 9. Creación de un ítem: Paso 2

Al hacer esto aparece una página en la que se deben rellenar los campos correspondientes con la información del "ítem". En la Figura 10 se muestra un ejemplo que hace referencia a un "ítem" del equipo Atlas que se ocupa de recoger información acerca del "OID" del equipo en cuestión. En ella destacan los campos "Nombre", con el nombre con el que se quiera identificar al ítem en cuestión, "Tipo", donde se especifica el tipo de agente que llevará a cabo la comunicación con el equipo, "Monitor", que es una clave identificativa dentro de la plataforma, el "SNMP OID" y la "Comunidad SNMP", en el caso de que el equipo se comunique vía SNMP, el "Puerto", que en el caso de ir en blanco se utilizará el puerto por defecto 161 y el "Tipo de información", la cual puede ser de tipo numérica o de carácter.

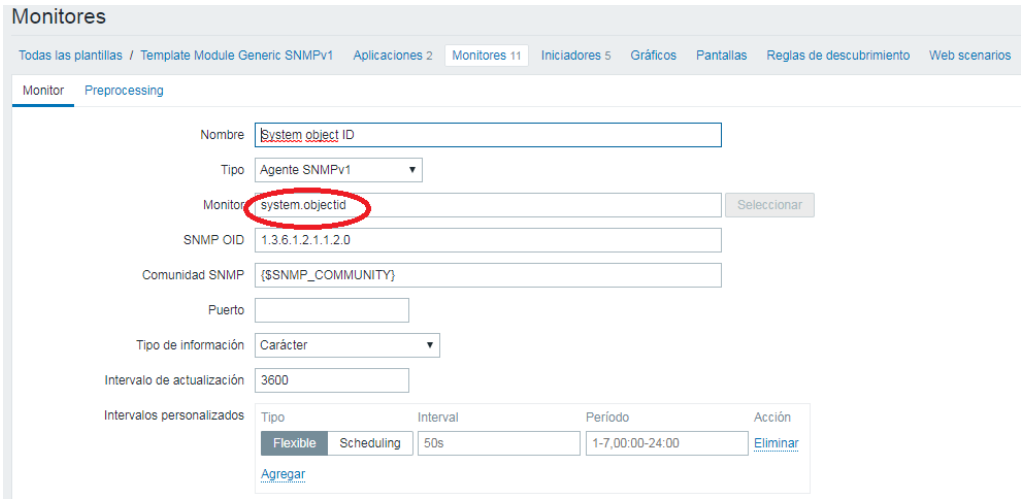


Figura 10. Creación de un ítem: Paso 3

2.4.3 Creación de "triggers"

Los "triggers" (o iniciadores según la traducción de Zabbix) son expresiones lógicas que "evalúan" los datos recopilados por los "ítems" y representan el estado actual del sistema. Aunque los "ítems" se utilicen para recopilar datos del sistema, es poco práctico tener que seguir continuamente la evolución de estos a la espera de un valor inusual o que sobrepase un umbral previamente definido. Por ello, se emplean los llamados "triggers" o disparadores que permiten definir un umbral que indique si el estado de los datos es "aceptable" o, por el contrario, si este valor se sobrepasa y pasa a ser considerado por Zabbix como un "Problema".

Para configurar un determinado "trigger" se debe ir a la pestaña "Configuración" y dentro de "Equipos" clicar en la opción "Iniciadores" "Crear iniciador" como se puede observar en las figuras Figura 11 y Figura 12.

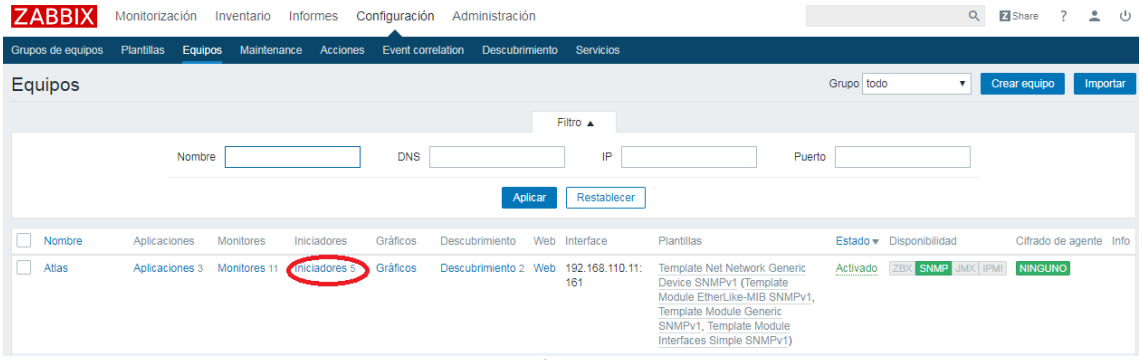


Figura 11. Creación de un trigger: Paso 1

Figura 12. Creación de un trigger: Paso 2

Al hacerlo, se ve una pantalla en la cual se rellenan los campos correspondientes con los valores adecuados para el “trigger”. En la Figura 13 se detalla un ejemplo para un “trigger” perteneciente a uno de los equipos que componen la red y para el cual se define una expresión que convierte en problemas los valores de carga en el procesador que se encuentren por encima del umbral marcado. Los campos más importantes o imprescindibles a la hora de crear un “trigger” son el campo “Nombre”, que dará una idea del desencadenante del disparador, “Gravedad”, donde se especificará entre una escala que va de “No clasificada” a “Crítica” la gravedad del desencadenante que propicia el disparador, y, finalmente, el campo “Expresión”, en el cual se definirá una expresión regular que Zabbix sea capaz de traducir e interpretar.

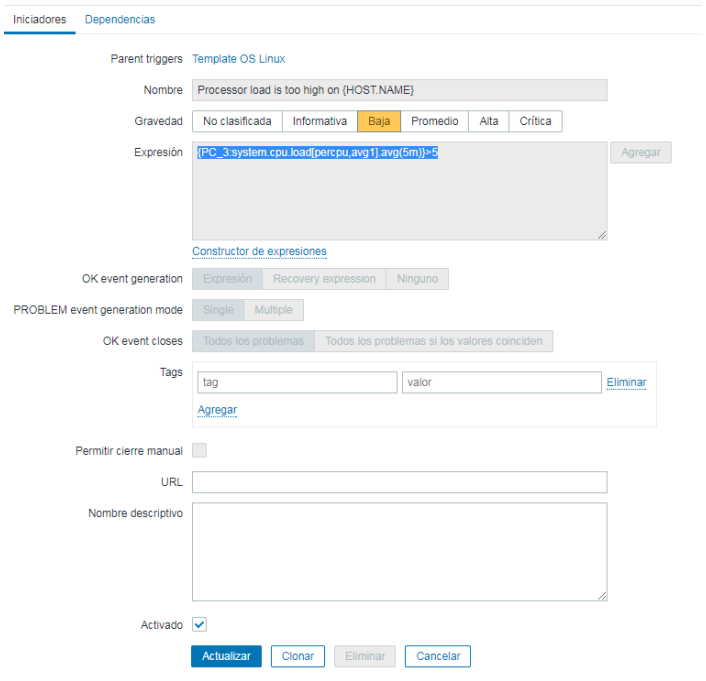


Figura 13. Creación de un trigger: Paso 3

2.4.4 Creación de gráficos

Los gráficos aportan una representación visual de los datos recogidos gracias a los "ítems" de cada equipo. Además, permiten correlacionar problemas y son totalmente personalizables.

Para crear un gráfico, se selecciona la pestaña "Equipos" del menú "Configuración" seleccionando la casilla "Crear Gráfico" en la parte derecha de la pantalla. Al hacerlo, se introduce un nombre para la gráfica en cuestión, el tipo "normal" y se activan los recuadros "Mostrar tiempo de trabajo" y "Mostrar iniciadores".

Finalmente se presiona en el botón "Agregar" y se añade el "ítem" del que se obtiene la información para componer la gráfica.

Un ejemplo de una gráfica puede ser el mostrado en la Figura 14, en donde se puede analizar el nivel de carga de la "CPU", observando que el nivel de carga medio del procesador en un minuto (ítem verde), sobrepasa el umbral establecido de valor cinco dos veces. Este caso se ha realizado intencionadamente provocando múltiples operaciones en el equipo para sobrecargarlo y ver así un caso extremo que reporta un problema a Zabbix.

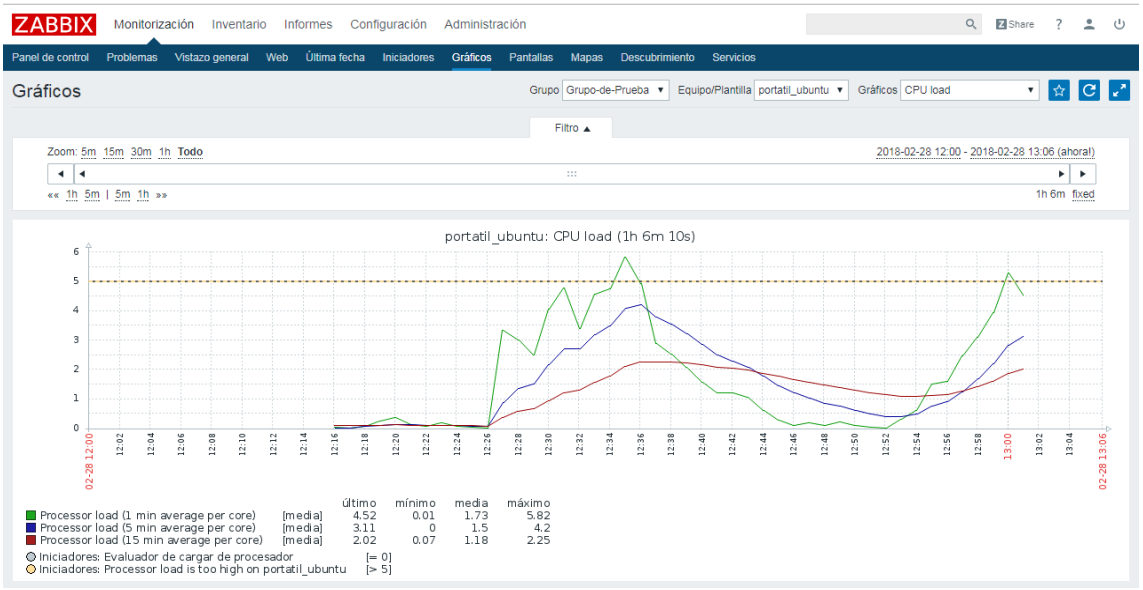


Figura 14. Creación de gráficos: Ejemplo de gráfico de la carga de CPU

2.4.5 Creación de mapas

Los mapas en Zabbix permiten obtener una visión global de la infraestructura Monitorizada. Estos mapas pueden ser públicos (abiertos a cualquier usuario) o privados (sólo determinados usuarios pueden acceder a ellos).

La configuración de un mapa en Zabbix requiere que primero se cree un mapa definiendo sus parámetros generales y luego se comience a llenar el mapa real con elementos y sus enlaces.

El mapa se puede cuenta con diferentes elementos, tales como un equipo o grupo de equipos, un "trigger", una imagen u otro mapa.

Los iconos se utilizan para representar elementos del mapa, pudiendo definir la información que se muestra en estos y destacar de forma visual las incidencias que surjan en ellos. Se puede, además, vincular los iconos y definir la información que se mostrará en los enlaces.

Los mapas se gestionan dentro del menú “Monitorización”, en la pestaña “mapas. Para su creación se debe ir a la sección “Crear mapa” en la parte derecha de la pantalla. Al hacerlo, se muestra una pantalla como la de la Figura 15, donde se introducen los datos correspondientes al mapa de la red (“Nombre”, “Anchura” y “Altura”) para, seguidamente, proceder a editarlo.

Para agregar un elemento, se hace clic en “Agregar” junto a “Icono”. El nuevo elemento aparece en la esquina superior izquierda del mapa. Seguidamente se hace clic en ese elemento para desplegar un menú como el mostrado en la XXX, donde se puede seleccionar la imagen acorde con el tipo de equipo que se quiere representar, así como su nombre y demás parámetros.

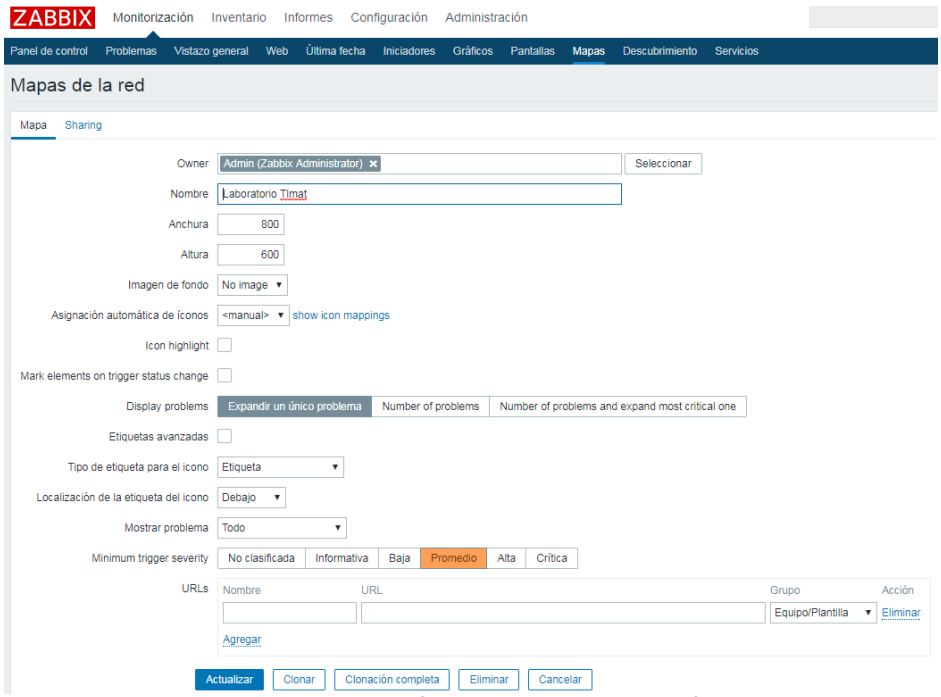
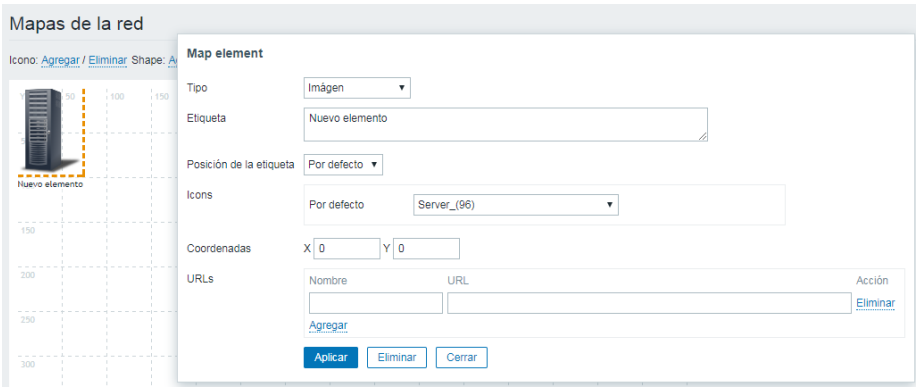


Figura 15. Creación de mapas: Configuración



2.4.6 Plantillas

Una plantilla es un conjunto de entidades que se pueden aplicar de forma conveniente a múltiples hosts.

Las plantillas se basan en el concepto de la similitud de los equipos en la red por lo que se reduce un trabajo innecesario de añadir entidades a mano a un equipo si se sabe que el equipo en cuestión comparte características con otro. De esta forma si se tiene un número elevado de equipos que comparten características se ahorra tiempo creando una plantilla específica y aplicándola a esos equipos antes que introducir las entidades correspondientes una a una en los susodichos equipos.

2.4.7 Creación de "dashboards"

En los "dashboards" se recoge cualquier información que se quiera mostrar del equipo en cuestión. Pero en este caso se da una visión mucho más completa del equipo, mostrando no sólo gráficas sino cualquier información trascendente que se quiera puntualizar del equipo.

Para crear un "dashboard" en la sección "monitoring", apartado "Dashboard" se hace clic en "Crear Dashboard". Al hacerlo se le asigna un propietario y un nombre y se entra en la pantalla de configuración en donde se añade contenido mediante la pestaña "añadir widget".

A continuación, se entra en un menú, Figura 16, el cual permite escoger el tipo de "widget" que se quiere añadir, así como asignarle un nombre, un intervalo de actualización y el número de datos que se desea que se muestren. Son muchos los tipos de "widgets" disponibles, por lo que, seguidamente, se realiza una breve descripción de cada uno:

- Action log: En este "widget" se muestran detalles de las operaciones de "acción" (notificaciones, comandos remotos, etc.).
- Clock: Muestra el tiempo que el servidor o host especificado lleva activo.
- Data overview: Este "widget" muestra los últimos datos obtenidos para un grupo de hosts, así como información de supervisión para dar una descripción general.
- Discovery status: En él se muestra un resumen del estado de las reglas de descubrimiento de red activas.
- Favourite graphs: Contiene accesos directos a los gráficos más importantes.
- Favourite maps: Contiene accesos directos a los mapas más importantes.
- Favourite screens: Este "widget" contiene accesos directos a las pantallas y presentaciones de diapositivas más necesarias.
- Graph: En este "widget" se puede mostrar un gráfico personalizado.
- Host status: Muestra información de alto nivel sobre la disponibilidad del host.
- Map: Muestra un mapa de red previamente configurado.
- Map navigation tree: Este "widget" permite construir una jerarquía de mapas que muestre las estadísticas de los problemas surgidos en cada mapa.
- Plain text: Con el "widget" de texto plano se muestran los últimos datos proporcionados por los ítems.
- Problems: Muestra los problemas que surgen en los equipos.
- Status of Zabbix: Permite visualizar información de alto nivel del servidor Zabbix.
- System status: Muestra el estado del sistema.
- Trigger overview: Permite visualizar los estados de los triggers para un grupo de hosts.
- URL: Muestra contenido URL de un recurso externo.

- Web monitoring: Muestra un resumen de los escenarios de monitorización activos.

Una vez añadido el widget elegido, se puede situar en la posición y tamaño que se desee dentro de la pantalla principal.

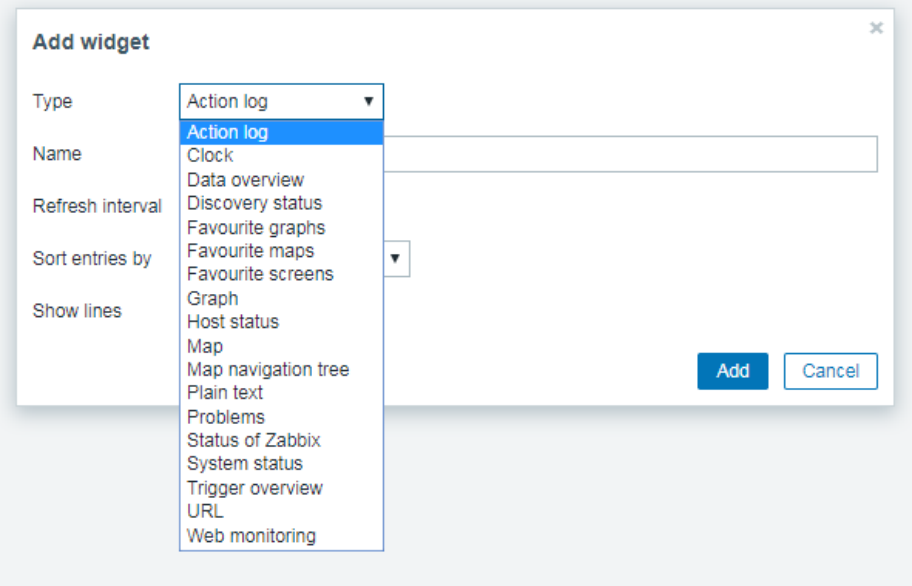


Figura 16. Menú de un widget

CAPÍTULO 3. ASPECTOS PRÁCTICOS

3.1 Creación de máquinas virtuales

La virtualización es el proceso por el cual se puede disponer de varios “ordenadores ficticios” dentro de un mismo ordenador físico. Para entender este proceso se deben tener claros conceptos como:

Máquina real, la cual se identifica como el ordenador en sí mismo, todos los componentes físicos que se pueden tocar, incluyendo CPU, pantalla, teclado, etc.

El sistema operativo anfitrión, es decir, el que se encuentra instalado en el ordenador o máquina real, que puede ser Windows, Linux, Mac, etc. En este caso se corresponde con la versión Windows 7 de la que disponen los ordenadores del laboratorio.

El software de virtualización que se elija, que es un programa informático instalado en el sistema operativo anfitrión y que sirve de contenedor de máquinas virtuales. Los más comunes suelen ser Virtual Box y VMWare. Este último, elegido para la realización del trabajo.

Por último, el sistema operativo virtual, denominado invitado que permite instalar y ejecutar programas en él como si fuese el sistema operativo anfitrión.

Entre las ventajas de implementar un sistema de virtualización se encuentran, el hecho de poder tener varios sistemas operativos ejecutándose de forma simultánea, la reducción de costes y de espacio físico. Entre los inconvenientes, las máquinas virtuales se encuentran limitadas por el hardware físico del ordenador, el rendimiento disminuye y la avería del sistema operativo anfitrión afecta a todas las máquinas virtuales alojadas en él. [20]

Por lo tanto, y para no alterar o modificar el comportamiento de los equipos del laboratorio, permitiendo así su uso continuado por los alumnos de la universidad durante la realización del proyecto, se trabaja con máquinas virtuales alojadas en los ordenadores personales “Local3”, “Local4” y “Local5”. En estas máquinas, se instalan entornos bajo sistemas operativos Ubuntu 16.04 a través de la herramienta gratuita VMWare Player.

La característica principal que deben tener las máquinas virtuales que se instalan en el laboratorio consiste en el empleo del modo de red Bridge, que consiste en conectar el sistema virtualizado directamente sobre la red física para permitir así la comunicación entre el nodo gestor y los nodos gestionados. En el modo NAT, sin embargo, los ordenadores se encontrarían conectados a la red tras la tarjeta de red física del sistema “Host”, lo que sería conveniente en el caso de disponer de una conexión a Internet mediante un módem u otro dispositivo que no tenga funciones de enrutamiento o NAT, el cual no es el caso del laboratorio docente. [21]

3.2 Descripción de la topología del laboratorio

El grupo de telemática cuenta con dos laboratorios de docencia, uno principal y otro secundario, interconectados entre sí. Para la realización de este proyecto se tendrán en consideración únicamente los equipos del laboratorio principal, al ser este el que soporta un mayor uso por parte del alumnado.

En la Figura 17, se puede observar un esquema de la red a monitorizar compuesta por los equipos del laboratorio principal. Este esquema muestra como un router Cisco 2600 se conecta a un switch 10/100/1000, al que a su vez se encuentran conectados todos los ordenadores personales que se utilizan en esta implementación. Además, la salida a internet se produce por un servidor de nombre ATLAS que actúa como "Gateway" o puerta de enlace entre el switch y el exterior. Los equipos pertenecientes a este laboratorio de docencia se configuran sobre una subred privada 192.168.110.X.

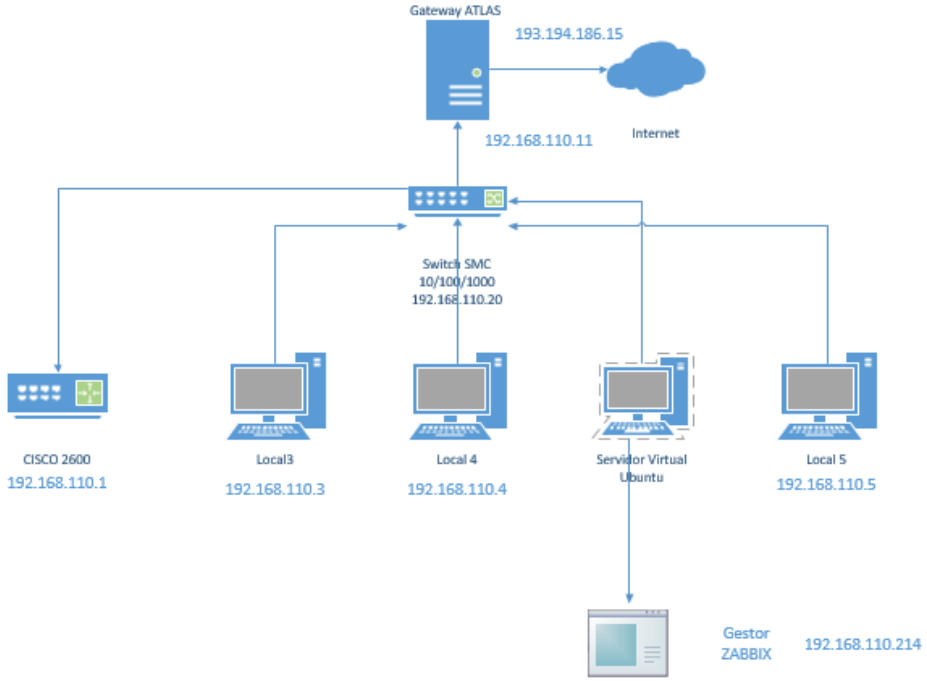


Figura 17. Esquema de red del laboratorio principal

El router Cisco 2600, que ocupa la primera dirección dentro de la subred, se encarga de interconectar los laboratorios entre sí y dispone de un agente SNMP propietario de Cisco, por lo que la comunicación entre Zabbix y el propio router se realiza mediante el protocolo SNMP.

El switch cuenta con 24 puertos, en los cuales se conectan los diez PC's de los que dispone actualmente el laboratorio. De entre esos diez ordenadores personales, para este proyecto se utilizarán los tres, nombrados del "Local3" al "Local5". Sus direcciones dentro de la subred se corresponden con el número del equipo (por ejemplo: el equipo "Local3" dispone de la dirección 192.168.110.3, el "Local4" de la dirección 192.168.110.4, etc.). Estos equipos funcionan bajo el sistema operativo Windows 7 x64 y para su monitorización se emplearán tanto agentes SNMP como agentes propios de Zabbix.

La instalación de los agentes en los ordenadores personales se realiza sobre máquinas virtuales, bajo la plataforma VMWare Player, en el caso de que se quiera utilizar el sistema operativo Linux. La versión utilizada en este caso es Ubuntu 16.04 LTS.

Cabe, además, destacar que la aplicación gestora se instala sobre un servidor virtual Ubuntu alojado en el equipo "Local5".

El servidor de nombre ATLAS, permite la salida a internet con la dirección IP pública 192.168.110.11. Este servidor realiza, además, funciones de firewall y NAT entre la red privada e Internet.

Para conocer elementos de interés que sean monitorizables se hace uso de una herramienta de apoyo conocida como MG-SOFT MIB Browser que permite navegar entre las mibs correspondientes a los equipos.

En el caso particular del servidor ATLAS fue necesario modificar su archivo de configuración para habilitar la visualización de parte de su mib, que es privada.

A la hora de hablar de las MIB's que se utilizan para obtener los parámetros monitorizables de los dispositivos de la red, destacan la RFC1213, de la que se obtienen datos como la descripción completa del sistema, su versión, hardware, sistema operativo y el tiempo que lleva activo el equipo desde su última re inicialización. Y la IF-MIB, la cual arroja datos como el número de interfaces de red y su descripción, la velocidad de la interfaz, así como el número de octetos de en entrada y de salida en cada segundo.

3.3 Instalación de la Plataforma de Gestión

Este apartado, se estructura en tres puntos fundamentales. Una primera parte, en la que se presenta la plataforma gestora Zabbix, especificando los pasos mediante los cuales se lleva a cabo su instalación.

En segundo lugar, se explica el modo en que han de instalarse los agentes nativos de Zabbix dependiendo de si el equipo anfitrión posee el sistema operativo Windows o Linux.

Finalmente, se termina con un tercer apartado en el que se da información acerca de la activación del servicio SNMP para realizar una monitorización "agentless" en los equipos del laboratorio.

3.3.1 Gestor Zabbix

La versión de Zabbix escogida para llevar a cabo la gestión del laboratorio será la 3.4 debido a su actualidad y buen funcionamiento.

Para instalar Zabbix existen requisitos de memoria y almacenamiento, que dependen de la cantidad de equipos y parámetros que se van a monitorizar, así como su intervalo de recolección.

Name	Platform	CPU/Memory	Database	Monitored hosts
Small	CentOS	Virtual Appliance	MySQL InnoDB	100
Medium	CentOS	2 CPU cores/2GB	MySQL InnoDB	500
Large	RedHat Enterprise Linux	4 CPU cores/8GB	RAID10 MySQL InnoDB or PostgreSQL	>1000
Very large	RedHat Enterprise Linux	8 CPU cores/16GB	Fast RAID10 MySQL InnoDB or PostgreSQL	>10000

Tabla 2. Requerimientos de hardware

En la Tabla 2, se recogen las necesidades medias para varios tipos de entornos de monitorización, desde pequeñas infraestructuras hasta las más grandes y complejas. Para este proyecto se toman como ejemplo las indicaciones dadas para un entorno pequeño, ya que el número de equipos que monitorizaremos es inferior a cien. Se emplea, además, una plataforma tipo Ubuntu y no una CentOS como se indica en la Tabla 2, debido a que se encontró mayor información sobre la primera durante la documentación de la instalación de Zabbix.

A la hora de calcular el espacio requerido en el disco para un sistema Zabbix como el implementado en este proyecto se deben tener en cuenta los siguientes parámetros:

- Configuración de Zabbix: Tamaño fijo, normalmente 10 MB
- Historial: días*(elementos/frecuencia de actualización)*24*3600*bytes
- Tendencias: días*(elementos/3600)*24*3600*bytes
- Eventos: días*eventos*24*3600*bytes

Por lo tanto, el espacio de disco requerido se puede calcular como la suma de los anteriores parámetros.

Un ejemplo de una configuración tipo para un proyecto como el realizado en el laboratorio docente sería:

En primer lugar, se ha de estimar el número de valores procesados por segundo (número promedio de nuevos valores que el servidor Zabbix recibe cada segundo), el cual depende del número de ítems que se monitoricen en la plataforma. Para este ejemplo se estima un valor aproximado de 60 ítems con una ratio de refresco de 60 segundos. Con todo ello se obtiene un valor de 60/60 = 1.

Esto significa que se agrega un nuevo valore a la base de datos de Zabbix cada segundo.

Zabbix mantiene los valores durante un período de tiempo fijo, normalmente varias semanas o meses, por lo que cada nuevo valor requiere una cierta cantidad de espacio en disco. Así que, si se tiene en cuenta un período de tiempo correspondiente al de un cuatrimestre escolar, por ejemplo, y se estima una recepción de un valor por segundo, el número de valores totales será (122*24*3600)*1 = 10540800 o lo que es lo mismo, unos 10,5M de valores.

Según la base de datos utilizada, el tipo de valores recibidos (flotantes, enteros, etc.), el espacio en el disco necesario para mantener un solo valor pueda variar desde los 40 bytes a cientos de bytes. Normalmente se encuentra en torno a los 90 bytes por valor para los elementos numéricos. En este caso concreto 10,5M de valores requerirán 10,5M * 90 bytes = 0,9GB, aproximadamente 1GB de espacio en el disco.

A continuación, se ha de calcular el espacio que se requiere para mantener los datos de las tendencias por un tiempo determinado, que en este caso será de un año. De esta forma, para un valor de 60 ítems se requerirá 60*24*365*90 = 473 MB en un año.

Por último, se ha de estimar el número de eventos y el tamaño que estos van a suponer para el entorno. Cada evento Zabbix requiere aproximadamente 170 bytes de espacio en disco. Es difícil estimar el número de eventos generados por Zabbix diariamente, pero en el peor de los casos, podemos suponer que Zabbix genera un evento por segundo. Esto significa que, si se quiere guardar un año de eventos, se requerirá 1*365*24*3600*170 = 5,36GB.

De las anteriores deducciones se puede estimar un espacio en disco necesario de:

10MB (Configuración de Zabbix) + 1GB (Historial) + 473MB (Tendencias) + 5,36GB (Eventos) = 6,843GB, aproximadamente 7GB.

El espacio en disco no se usará inmediatamente después de la instalación de Zabbix. Sino que irá incrementándose a medida que la plataforma lo requiera.

Estos cálculos son orientativos y debido a que en este proyecto no se recopila información en la base de datos durante un período de tiempo continuado no son necesarios por lo que es suficiente con un espacio de disco estándar.

Antes de comenzar con la instalación de Zabbix, en el terminal gestor (“Local5”) sobre un servidor Ubuntu, se ha de realizar la instalación de unos requisitos previos. Estos requisitos son un servidor web apache junto con sus módulos para la base de datos MySQL y el lenguaje de código abierto PHP que se ejecuta en el servidor. Además, se instala el paquete “phpmyadmin”, que es una herramienta escrita en lenguaje PHP y se usa para manejar la administración de la base de datos MySQL a través de páginas web, utilizando internet.

- Paso 1: Instalación de apache

El servidor Web Apache es actualmente el más popular del mundo. Está bien documentado, y ha sido ampliamente utilizado en la historia de la web, lo que hace que sea una gran opción por defecto para montar un sitio web.

Además, se puede instalar Apache fácilmente desde el gestor de paquetes de Ubuntu, “apt”. Un gestor de paquetes permite instalar con mayor facilidad un software desde un repositorio mantenido por Ubuntu. Para instalarlo se escribe el siguiente comando:

```
#sudo apt-get –y install apache2
```

Al utilizar el comando “sudo”, esta operación es ejecutada con privilegios de administrador, por lo que se pedirá la contraseña asignada para dicho administrador en el servidor Ubuntu.

Una vez que se haya ingresado la contraseña,” apt” dirá qué paquetes planea instalar y cuánto espacio adicional ocuparán en el disco.

Finalmente, la instalación del servidor apache acabará con una instrucción de reinicio:

```
#sudo systemctl restart apache2
```

- Paso 2: Instalación de MySQL

Ahora que ya se tiene el servidor web configurado y corriendo, es el momento de instalar MySQL. MySQL es un sistema de gestión de base de datos. Básicamente, se encarga de organizar y facilitar el acceso a las bases de datos donde almacenar la información.

Una vez más, se puede usar “apt” para adquirir e instalar el software. Esta vez, también se van a instalar otros paquetes "auxiliares" que permitirán a los componentes comunicarse unos con otros:

```
#sudo apt-get –y install mysql-server mysql-client
```

Durante la instalación, el servidor pedirá que se seleccione y se confirme una contraseña para el usuario "root" de MySQL. Esta es una cuenta administrativa en MySQL que ha aumentado privilegios. Para entenderlo, se puede pensar en ello como algo similar a la cuenta de root para el propio servidor (la que se está configurando ahora es una cuenta específica de MySQL). Conviene, además, que sea una contraseña segura y única. Como ejemplo se supone que se ha usado la contraseña “zabbix”.

Con el establecimiento de la contraseña el sistema de base de datos ya se encuentra configurado.

- Paso 3: Instalación de PHP

PHP es el componente de la configuración que procesará código para mostrar contenido dinámico. Además, puede ejecutar secuencias de comandos, conectarse a las bases de datos MySQL para obtener información y entregar el contenido procesado al servidor web para mostrarlo.

Como se hizo anteriormente, se puede aprovechar el sistema "apt" para instalar los componentes. Junto con PHP se van a incluir algunos paquetes de ayuda, de esta forma el código PHP se podrá ejecutar en el servidor apache y comunicarse con la base de datos MySQL:

```
#sudo apt-get -y install php7.0 libapache2-mod-php7.0
```

A continuación, se reinicia el servidor web apache para que se apliquen los cambios:

```
#sudo systemctl restart apache2
```

- Paso 4: Instalación de módulos adicionales de PHP

Para mejorar la funcionalidad de PHP, se pueden instalar opcionalmente algunos módulos adicionales. Para ello se ejecuta el siguiente comando:

```
#sudo apt-get install php7.0-mysql php7.0-mcrypt php7.0-curl php-all-
dev php7.0-gd php-pear php-imagick php7.0-pspell php7.0-xmlrpc php7.0-
mbstring -y
```

Posteriormente se reinicia el servidor:

```
#sudo systemctl restart apache2
```

Y se finaliza con la instalación del paquete "phpmyadmin":

```
#sudo apt-get install phpmyadmin php-mbstring php-gettext -y
```

Después de ejecutar el comando, se ha de seleccionar el tipo de servidor adecuado para realizar la configuración automática junto con "phpmyadmin". Es por ello que se elige la opción "apache2", como muestra la Figura 18, debido a que es el tipo de servidor web que se instaló previamente. [22]

A continuación, en la Figura 19, se puede observar cómo se plantea la posibilidad de realizar la configuración de la base de datos asociada a "phpmyadmin" de forma automática con "dbconfig-common" o, por el contrario, hacerla de forma manual en el caso de ser un administrador "avanzado". En este caso se selecciona la respuesta "Yes" para que la configuración se realice de manera automática.

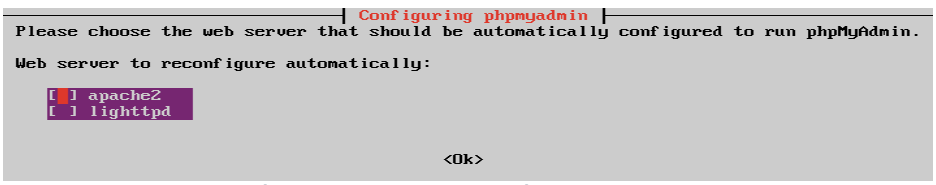


Figura 18. Instalación phpmyadmin: Selección de apache2 como servidor web

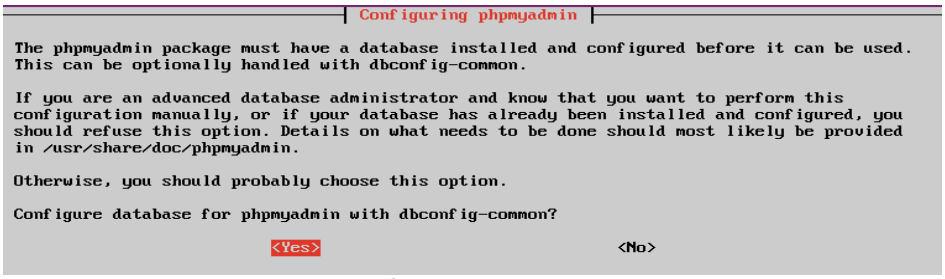


Figura 19. Instalación phpmyadmin: Dbconfig-common

Para finalizar la instalación de los requisitos previos, se ingresa la contraseña que se pide ("zabbix") y que es la correspondiente a la aplicación MySQL para phpmyadmin.

Una vez dichos requisitos sean instalados se procede a la obtención e instalación de Zabbix ejecutando los siguientes comandos:

- Paso 1: Instalación del servidor Zabbix

```
#wget    http://repo.zabbix.com/zabbix/3.4/ubuntu/pool/main/z/zabbix-release/zabbix-release_3.4-1+xenial_all.deb

#sudo dpkg -i zabbix-release_3.4-1+xenial_all.deb

#sudo apt-get update

#sudo apt-get install zabbix-server-mysql zabbix-frontend-php php7.0-mbstring php7.0-bcmath php7.0-xml curl
```

- Paso 2: Creación e importación de la base de datos

Una vez obtenida e instalada la plataforma, se crea una base de datos del tipo MySQL ejecutando los siguientes comandos:

```
#mysql -u root -p

mysql > create database zabbix character set utf8 collate utf8_bin;

mysql > grant all privileges on zabbix.* to zabbix@localhost identified by '****';

mysql > quit;

#zcat  /usr/share/doc/zabbix-server-mysql/create.sql.gz  |  mysql  -u zabbix -p zabbix
```

Con ello se consigue crear una base de datos asociada a Zabbix y otorgarla privilegios para que pueda trabajar y comunicarse con la plataforma.

El comando "zcat" permite realizar la importación de la base de datos.

- Paso 3: Configuración de la base de datos para Zabbix

Para adaptar la configuración de la base de datos a la infraestructura de la red del laboratorio sobre Zabbix se deben editar los parámetros "DBHost", "DBName", "DBUser" y "DBPassword" del archivo de configuración "zabbix_server.conf". Para ello:

```
#sudo nano /etc/zabbix/zabbix_server.conf
```

Se introducen los parámetros de la Figura 20, que se corresponden con el campo “DBHost” que debe coincidir con el “hostname” de la base de datos. En este caso, para una base de datos MySQL, el nombre correspondiente es “localhost”.

```
### Option: DBHost
#       Database host name.
#       If set to localhost, socket is used for MySQL.
#       If set to empty string, socket is used for PostgreSQL.
#
# Mandatory: no
# Default:
DBHost=localhost
```

Figura 20. Archivo de configuración de la base de datos de Zabbix: DBHost

Para el campo “DBName” mostrado en la Figura 21 y correspondiente al nombre que se le dio anteriormente a la base de datos asociada con Zabbix se introduce “zabbix”.

```
### Option: DBName
#       Database name.
#       For SQLite3 path to database file must be provided. DBUser and DBPassword are ignored.
#
# Mandatory: yes
# Default:
# DBName=

DBName=zabbix
```

Figura 21. Archivo de configuración de la base de datos de Zabbix: DBName

En la Figura 22 Figura 21, se muestra el nombre de usuario “zabbix” correspondiente a la base de datos.

```
### Option: DBUser
#       Database user. Ignored for SQLite.
#
# Mandatory: no
# Default:
# DBUser=

DBUser=zabbix
```

Figura 22. Archivo de configuración de la base de datos de Zabbix: DBUser

Y, finalmente, se especifica la contraseña para la base de datos, que se ha de corresponder con la contraseña “zabbix” asignada anteriormente.

- Paso 4: Arranque del servidor Zabbix

A continuación, se inicia el servidor Zabbix instalado previamente.

```
#sudo service zabbix-server start

#sudo update-rc.d zabbix-server enable
```

- Paso 5: Configuración de PHP para Zabbix

También se debe establecer la configuración de PHP para Zabbix. Para ello, se modifican los siguientes parámetros del archivo “php.ini”:

```
#sudo nano /etc/php/7.0/apache2/php.ini
```

El primer valor a modificar es el “post_max_size” que establece un límite en el tamaño máximo de los datos que PHP acepte. Como muestra la Figura 23, este valor será de 16M.

```
; Maximum size of POST data that PHP will accept.
; Its value may be 0 to disable the limit. It is ignored if POST data reading
; is disabled through enable_post_data_reading.
; http://php.net/post-max-size
post_max_size = 16M
```

Figura 23. Archivo de configuración de PHP: Post_max_size

A continuación, los tiempos máximos de ejecución y entrada se configuran a un valor igual a 300, como se observa en las figuras Figura 24 y Figura 25.

Por último, se establece el valor de la zona horaria situándolo en “Europe/Madrid”.

```
;;;;;;;;;;;;;;;;;;;
; Resource Limits ;
;;;;;;;;;;;;;;;;;;;

; Maximum execution time of each script, in seconds
; http://php.net/max-execution-time
; Note: This directive is hardcoded to 0 for the CLI SAPI
max_execution_time = 300_
```

Figura 24. Archivo de configuración de PHP: Max_execution_time

```
; Maximum amount of time each script may spend parsing request data. It's a good
; idea to limit this time on productions servers in order to eliminate unexpectedly
; long running scripts.
; Note: This directive is hardcoded to -1 for the CLI SAPI
; Default Value: -1 (Unlimited)
; Development Value: 60 (60 seconds)
; Production Value: 60 (60 seconds)
; http://php.net/max-input-time
max_input_time = 300
```

Figura 25. Archivo de configuración de PHP: Max_input_time

Una vez realizados los pasos anteriores se lleva a cabo la configuración del entorno web de Zabbix. Para ello, en un navegador cualquiera como puede ser “Chrome” de Google, se ingresa la siguiente URL: http://”IP Pública del Servidor”/zabbix.

La estación gestora elegida, tendrá la dirección IP 192.168.110.206. A continuación, se siguen los pasos para realizar la configuración web.

En primer lugar, la interfaz web de Zabbix muestra un chequeo o comprobación del estado de todos los prerrequisitos necesarios para el buen funcionamiento de la plataforma Zabbix. La Figura 26 muestra como todos ellos recogen el estado “OK”.

Seguidamente se procede a la confirmación de la configuración anteriormente realizada para la base de datos. En ella, únicamente se deberá rellenar el campo “Password” con la contraseña “zabbix”, como muestra la Figura 27. Los demás parámetros se corresponderán con los configurados para la base de datos en el fichero “zabbix_server.conf”.

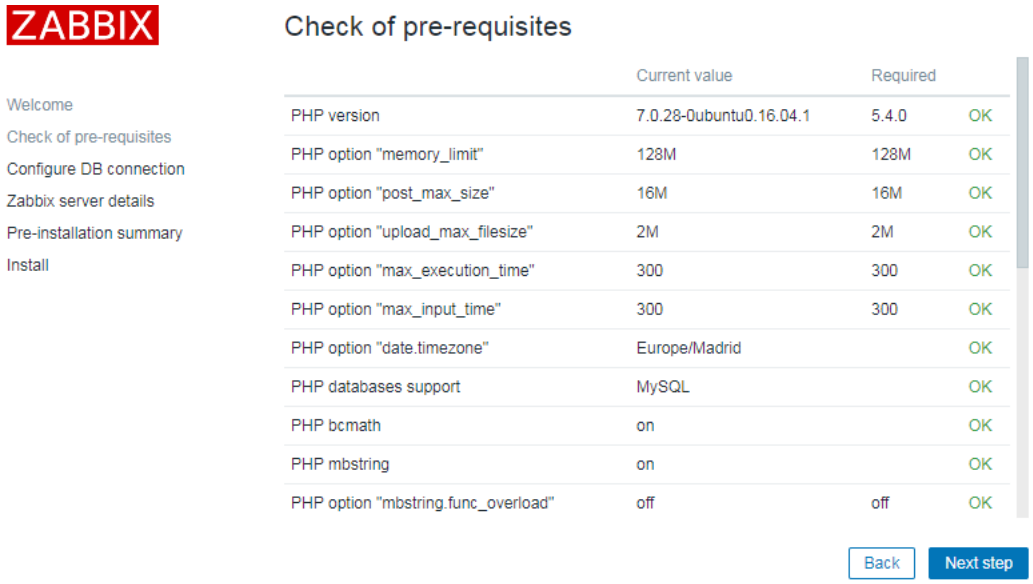


Figura 26. Configuración de la interfaz web de Zabbix: Prerrequisitos

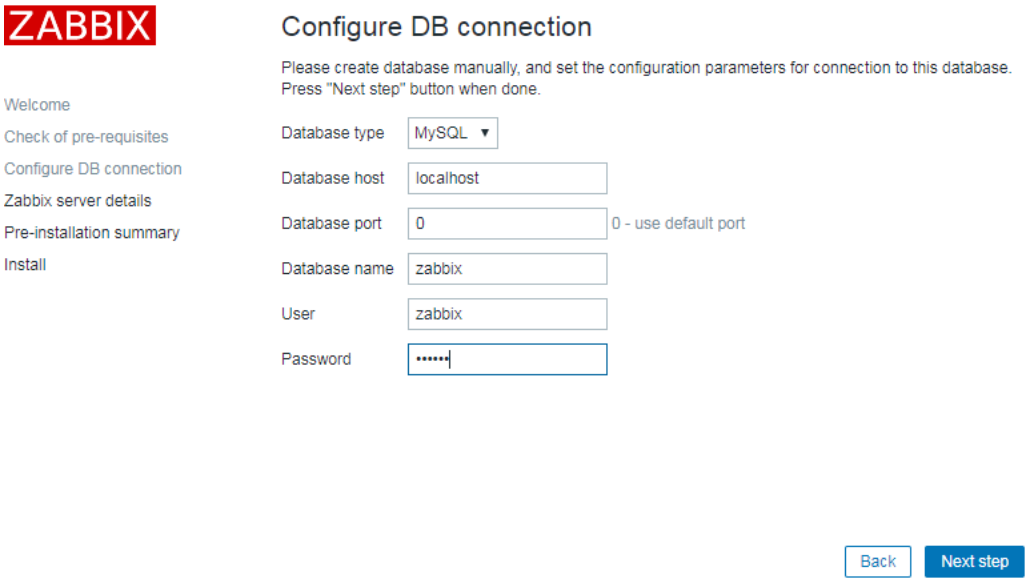


Figura 27. Configuración de la interfaz web de Zabbix: Conexión de la base de datos

Además, se dejan por defecto los parámetros “Host”, “Port” y “Name” del servidor Zabbix, Figura 28.

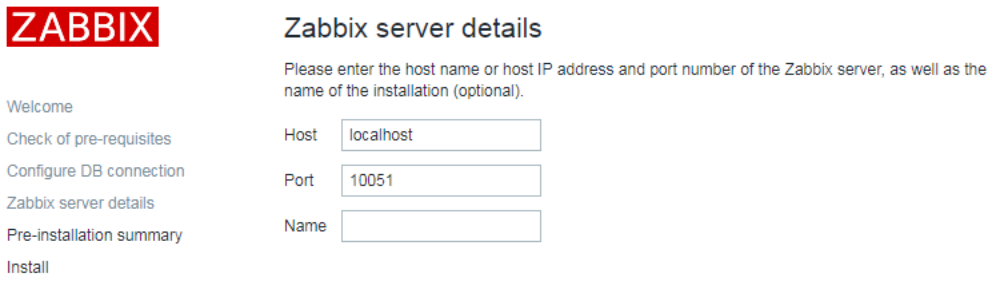


Figura 28. Configuración de la interfaz web de Zabbix: Detalles del servidor

Una vez seguidos todos los pasos anteriores se accede a la pantalla principal de la interfaz. Para ello se introduce el usuario “Admin” y contraseña “zabbix”. Ambos son los valores por defecto para ingresar en Zabbix como administrador.

3.3.2 Agentes Zabbix

Antes de comenzar con la configuración de los equipos en la interfaz web, estos deben disponer de los agentes necesarios para que se establezca la comunicación entre ellos y la plataforma. En este apartado se explica el funcionamiento e instalación de los agentes que Zabbix proporciona de forma nativa dependiendo del sistema operativo empleado.

Son dos las opciones que se plantean, instalar el agente sobre el sistema operativo Windows 7 con el que cuentan los ordenadores personales del laboratorio o instalar ese agente nativo de zabbix sobre el sistema operativo Ubuntu alojado en una máquina virtual.

- Opción 1: Instalar un agente Zabbix en Windows

Para llevar a cabo la instalación de un agente Zabbix en una máquina con un sistema operativo Windows se debe descargar en primer lugar el software agente de la página https://www.zabbix.com/download_agents, en este caso el correspondiente a la versión 3.4.

A continuación, se crea una carpeta en la unidad [C:] del equipo, de nombre "zabbix". Allí se guarda el archivo previamente descargado del sitio web y se descomprime. Una vez hecho esto, se edita el archivo "zabbix_agentd.win.conf" para cambiar los parámetros siguientes:

```
Server=″ IP pública del servidor Zabbix″

ListenIP=″ Dirección del equipo donde se está instalando el agente
Zabbix″

ServerActive=″ IP pública del servidor Zabbix″

Hostname=″ Nombre del equipo en el que se instala el agente″
```

Por último, se guardan los cambios y se instala el servicio Zabbix desde un terminal del sistema o "CMD" con los comandos:

Con la opción –c se copia el archivo descargado del sitio web a la carpeta correspondiente.

```
#C:\zabbix\bin\win64\zabbix_agentd.exe –c

#C:\zabbix\conf\zabbix_agentd.win.conf –i
```

Finalmente, con la opción –s se inicia el servicio y se concluye la instalación.

```
#C:\zabbix\bin\win64\zabbix_agentd.exe –s
```

- Opción 2: Instalar un agente Zabbix en Ubuntu

Por otro lado, para realizar la instalación de un agente Zabbix en una máquina virtual con el sistema operativo Ubuntu, versión 16.04 LTS, se debe introducir el siguiente comando por consola:

```
#sudo apt install zabbix-agent
```

Y editar el fichero de configuración “zabbix_agentd.conf” para realizar los cambios pertinentes:

```
#sudo nano /etc/zabbix/zabbix_agentd.conf
```

En la Figura 29 se observa la configuración del campo “Server” en el que se introduce la dirección IP del servidor Zabbix instalado, que en este caso se corresponde con la dirección 192.168.110.206.

```
### Option: Server
#       List of comma delimited IP addresses (or hostnames) of Zabbix servers.
#       Incoming connections will be accepted only from the hosts listed here.
#       If IPv6 support is enabled then '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' are treated equally.
#
# Mandatory: no
# Default:
# Server=

Server=192.168.110.206
```

Figura 29. Archivo de configuración del agente Zabbix para Ubuntu Server

A continuación, en la Figura 30, se establece la dirección IP de la máquina virtual en la cual se está instalando el agente Zabbix.

```
### Option: ListenIP
#       List of comma delimited IP addresses that the agent should listen on.
#       First IP address is sent to Zabbix server if connecting to it to retrieve list of active checks.
#
# Mandatory: no
# Default:
ListenIP=192.168.110.128
```

Figura 30. Archivo de configuración del agente Zabbix para Ubuntu: ListenIP

Además, como muestra el campo “ServerActive” de la Figura 31, se repite la dirección IP del servidor Zabbix.

```
### Option: ServerActive
#       List of comma delimited IP:port (or hostname:port) pairs of Zabbix servers for active checks.
#       If port is not specified, default port is used.
#       IPv6 addresses must be enclosed in square brackets if port for that host is specified.
#       If port is not specified, square brackets for IPv6 addresses are optional.
#       If this parameter is not specified, active checks are disabled.
#       Example: ServerActive=127.0.0.1:20051,zabbix.domain,[::1]:30051,::1,[12fc::1]
#
# Mandatory: no
# Default:
# ServerActive=

ServerActive=192.168.110.206
```

Figura 31. Archivo de configuración del agente Zabbix para Ubuntu: ServerActive

Finalmente, en el campo “Hostname” se introduce el nombre de la máquina en la que se instala el agente. En este caso, Ubuntu, como muestra la Figura 32.

```
### Option: Hostname
#       Unique, case sensitive hostname.
#       Required for active checks and must match hostname as configured on the server.
#       Value is acquired from HostnameItem if undefined.
#
# Mandatory: no
# Default:
# Hostname=

Hostname=ubuntu
```

Figura 32. Archivo de configuración del agente Zabbix para Ubuntu: Hostname

Algunos de estos parámetros aparecen comentados por defecto, por lo que para que los cambios se apliquen correctamente se debe eliminar la almohadilla.

Por último, se ha de reiniciar el servicio “Zabbix Agent” con el comando:

```
#sudo service zabbix-agent restart
```

3.3.3 Servicio SNMP

En el caso de optar por realizar la comunicación entre el equipo y el servidor Zabbix mediante el protocolo SNMP, basta con activar este servicio sobre el sistema operativo Windows en el equipo.

A la hora de configurar el servicio de “Simple Network Management Protocol” o “SNMP” se ingresa en el panel de control, se selecciona “Herramientas administrativas” y, a continuación, se selecciona “Administración de equipos”.

En el árbol de la consola se expande “servicios y aplicaciones” para interactuar con la pestaña “Servicios”. Después, en el panel derecho se hace doble clic en “Servicio SNMP”, con lo que se despliega una ventana. En ella, se busca la ficha “Agente” y se escribe el nombre del usuario o administrador del equipo en el cuadro de contacto, así como la ubicación física del equipo en el cuadro ubicación.

Por último, en la ficha de generación de “traps”, se agrega además la comunidad “SNMP” junto con la IP correspondiente al servidor de Zabbix.

La activación de este servicio permitirá realizar una monitorización sin agentes en los equipos, obteniendo la información directamente de los paquetes de red que se transmiten entre sus componentes. El análisis de estos paquetes proporcionará, además, datos sobre el rendimiento y la disponibilidad del servicio.

CAPÍTULO 4. IMPLEMENTACIÓN EN EL LABORATORIO

En este capítulo se realiza una descripción detallada de cada uno de los equipos que conforman la red del laboratorio docente, junto con el proceso llevado a cabo para que la plataforma Zabbix los reconozca e intercambie información con ellos. Para ello, se determinan los ítems, triggers, gráficos, mapas y dashboards utilizados para llevar a cabo la monitorización.

En primer lugar, se añaden todos los equipos de la forma en la que se indicó en el apartado "2.4.1 Creación de hosts", especificando para cada uno su nombre e IP correspondientes como recoge la Tabla 3.

Nombre descriptivo del host	**Dirección IP**
Switch SMC	192.168.110.20
Atlas	192.168.110.11
Cisco 2600	192.168.110.1
PC1 (Local3)	192.168.110.225
PC2 (Local4)	192.168.110.211
PC3 (Local5)	192.168.110.227

Tabla 3. Nombre descriptivo y dirección IP

Las direcciones IP de los ordenadores personales que muestra la Tabla 3, se corresponden con las empleadas por las máquinas virtuales con el sistema operativo Ubuntu instaladas en estos. Finalmente, y después de muchas pruebas, se decidió que la utilización de máquinas virtuales era la forma óptima de trabajar, ya que los ordenadores del laboratorio docente se encuentran "congelados", es decir, que una vez se apagan vuelven al estado inicial sin guardar ningún cambio.

Una vez estos equipos se encuentren disponibles y activos para su uso a través de la interfaz web de Zabbix, se procede a asignarles unas plantillas con las que cuenta la plataforma por defecto y que permiten obtener de una forma sencilla los parámetros más básicos de los equipos.

Para los equipos monitorizados vía SNMP, como son el Switch, el servidor Atlas y el router Cisco 2600, se utiliza la plantilla "Template Net Network Generic Device SNMPv1". Esta plantilla proporciona información básica de cada uno de los equipos a través del protocolo SNMP versión 1 como el tiempo de actividad, la descripción del equipo y el tráfico de entrada y salida.

Por el contrario, para los ordenadores bajo el sistema operativo Ubuntu se hace uso de la plantilla "Template OS Linux", que permite obtener valores de la CPU como su carga, utilización, etc. Además del uso, estado y capacidad de la memoria del equipo.

Zabbix requiere que cualquier equipo que se cree en su interfaz esté asociado a un grupo de hosts. Por ello, se crea un grupo llamado “Laboratorio Tlmat” al que irán asociados todos los equipos del entorno de trabajo.

En este punto se procede a realizar una configuración más detallada de cada equipo, complementando las plantillas con la creación de ítems más específicos, gráficos que representen los datos obtenidos y dashboards que den al gestor de la red, en un solo instante, la información más interesante para este.

4.1 Switch SMC 10/100/1000

Un switch o conmutador es un dispositivo de interconexión de redes informáticas. Un conmutador interconecta dos o más partes de una red, funcionando como un puente que transmite datos de un segmento a otro.

El funcionamiento de un conmutador o switch tiene lugar porque el mismo tiene la capacidad de aprender y almacenar direcciones de red de dispositivos alcanzables a través de sus puertos. A diferencia de lo que ocurre con un hub o concentrador, el switch hace que la información dirigida a un dispositivo vaya desde un puerto origen a otro puerto destino. [23]

El switch del laboratorio consta de 24 puertos a los cuales se conectan los ordenadores personales. Por ello, y para complementar la plantilla utilizada por este equipo se han añadido los siguientes ítems a los proporcionados por la plantilla “Template Net Network Generic Device SNMPv1”, de la forma que indica el apartado “2.4.2 Creación de ítems”:

Dirección MAC del puerto número 5:

El hecho de que se elija este puerto en concreto se debe a su correspondencia con el equipo cinco, sobre el cual se encuentra instalado el servidor de Zabbix, y por lo que se considera el más importante. Es por ello que se cree conveniente configurar un ítem para que muestre su dirección MAC.

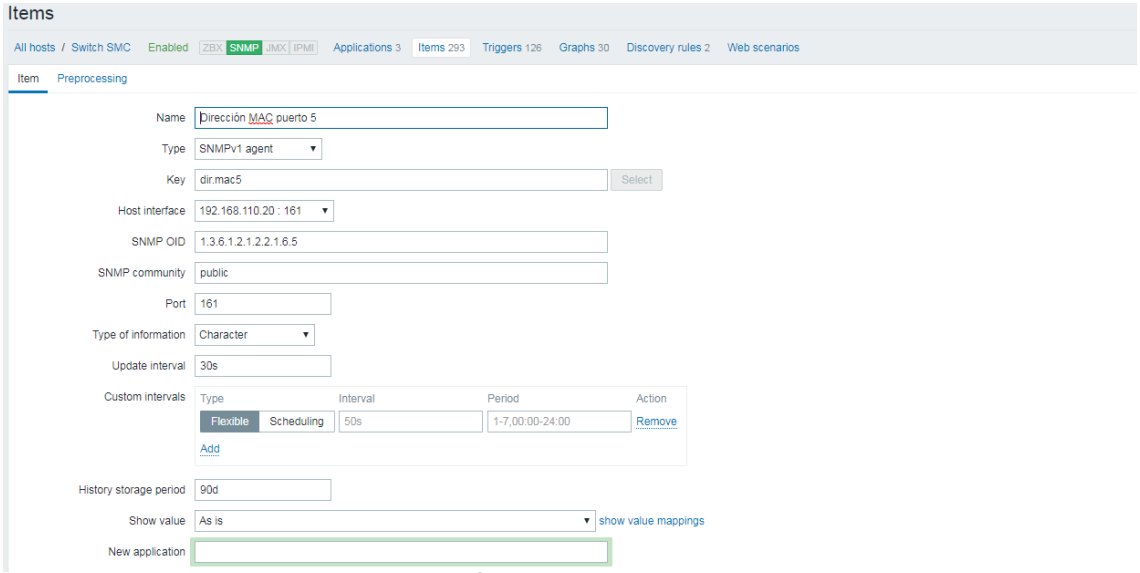


Figura 33. Ítem: Dirección MAC puerto 5

La creación de este ítem aparece reflejada en la **Error! Reference source not found.** y en la que destacan los siguientes campos:

- Name: Dirección MAC puerto 5

- Type: SNMPv1 agent
- Key: dir.mac5
- SNMP OID: 1.3.6.1.2.1.2.2.1.6.5
- SNMP Community: public
- Port: 161
- Type of information: Character

Dirección MAC del equipo:

- Name: MAC switch
- Type: SNMPv1 agent
- Key: dirección.mac
- SNMP OID: 1.3.6.1.4.1.202.20.67.1.5.6.1
- SNMP Community: public
- Port: 161
- Type of information:Character

Modelo:

- Name: modelo switch
- Type: SNMPv1 agent
- Key: modelo.switch
- SNMP OID: 1.3.6.1.4.1.202.20.67.1.1.5.1
- SNMP Community: public
- Port: 161
- Type of information: Character

A continuación, se configura una gráfica que muestre el tráfico de salida en el switch para cada puerto, y poder comparar así la actividad entre ellos.

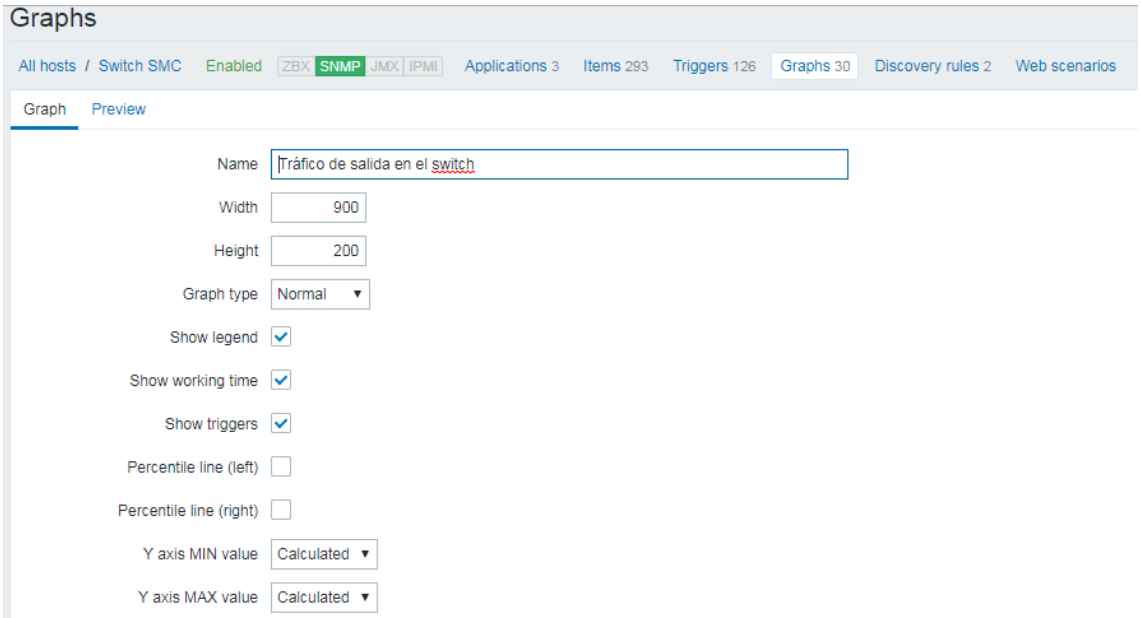


Figura 34. Gráfica switch: Configuración 1

En la Figura 34 se detallan los campos a completar para la creación de esta gráfica, como el nombre, las dimensiones (altura y anchura), si se quiere mostrar una leyenda o los triggers si los hubiera, etc. La configuración finaliza con la adición de los ítems de

los que se obtiene la información a representar. En este caso, el dato de los bits enviados a través de cada puerto de los 24 que componen el switch, Figura 35.

9:	Switch SMC: Interface Ethernet Port on unit 1, port 9: Bits sent	avg	Line	Left	F230E0 Remove
10:	Switch SMC: Interface Ethernet Port on unit 1, port 10: Bits sent	avg	Line	Left	5CCD18 Remove
11:	Switch SMC: Interface Ethernet Port on unit 1, port 11: Bits sent	avg	Line	Left	BB2A02 Remove
12:	Switch SMC: Interface Ethernet Port on unit 1, port 12: Bits sent	avg	Line	Left	5A2B57 Remove
13:	Switch SMC: Interface Ethernet Port on unit 1, port 13: Bits sent	avg	Line	Left	89ABF8 Remove
14:	Switch SMC: Interface Ethernet Port on unit 1, port 14: Bits sent	avg	Line	Left	7EC25C Remove
15:	Switch SMC: Interface Ethernet Port on unit 1, port 15: Bits sent	avg	Line	Left	274482 Remove
16:	Switch SMC: Interface Ethernet Port on unit 1, port 16: Bits sent	avg	Line	Left	2B5429 Remove
17:	Switch SMC: Interface Ethernet Port on unit 1, port 17: Bits sent	avg	Line	Left	8048B4 Remove
18:	Switch SMC: Interface Ethernet Port on unit 1, port 18: Bits sent	avg	Line	Left	FD5434 Remove
19:	Switch SMC: Interface Ethernet Port on unit 1, port 19: Bits sent	avg	Line	Left	790E1F Remove
20:	Switch SMC: Interface Ethernet Port on unit 1, port 20: Bits sent	avg	Line	Left	87AC4D Remove
21:	Switch SMC: Interface Ethernet Port on unit 1, port 21: Bits sent	avg	Line	Left	E89DF4 Remove
22:	Switch SMC: Interface Ethernet Port on unit 1, port 22: Bits sent	avg	Line	Left	1A7C11 Remove
23:	Switch SMC: Interface Ethernet Port on unit 1, port 23: Bits sent	avg	Line	Left	F63100 Remove
24:	Switch SMC: Interface Ethernet Port on unit 1, port 24: Bits sent	avg	Line	Left	2774A4 Remove

Add

Figura 35. Gráfica switch: Configuración 2

El resultado es una gráfica como la mostrada en la Figura 36.

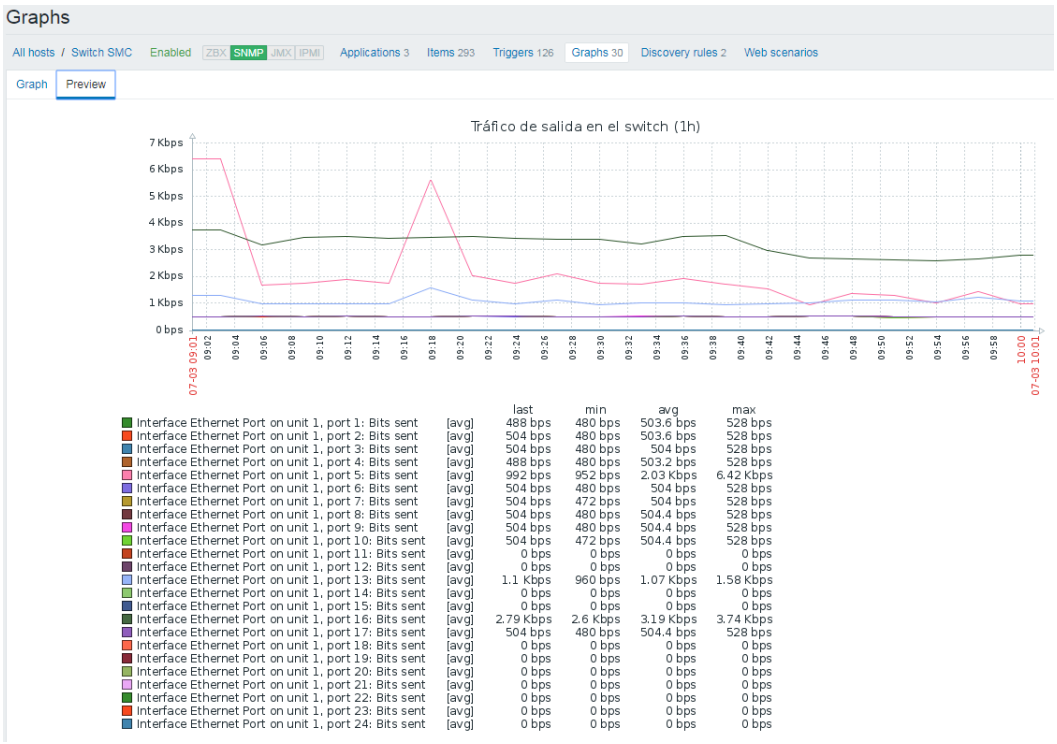


Figura 36. Gráfica switch: Tráfico de salida

En esta gráfica se pueden observar diferentes valores de tráfico diferenciados por colores (como muestra la leyenda) y en la que el puerto correspondiente al color rosa (puerto 5) llega en ciertos puntos a obtener el valor más alto de tráfico. Esto es debido a que en el momento de la realización de la captura sólo se trabajó con el equipo cinco, asociado al puerto cinco, manteniendo el resto de equipos en un nivel de trabajo bajo, para realizar de esta forma la comprobación de que la gráfica se adapta a la realidad. Además, esta gráfica puede llegar a tener una doble funcionalidad si uno se percata de que determinados puertos no transmiten tráfico de ningún tipo. Llegando así a la conclusión de que esos puertos se encuentran sin conexión alguna y están, por lo tanto, disponibles para agregar nuevos equipos.

Zabbix permite presentar la información recogida por los ítems y las gráficas de forma conjunta para tener una visión global de los parámetros monitorizados del equipo. El término con el que se conoce esta presentación dentro de la interfaz web de Zabbix es “Dashboard”.

Para el equipo switch del laboratorio se ha creado un tipo de dashboard como el que muestran las figuras Figura 37 y Figura 38.

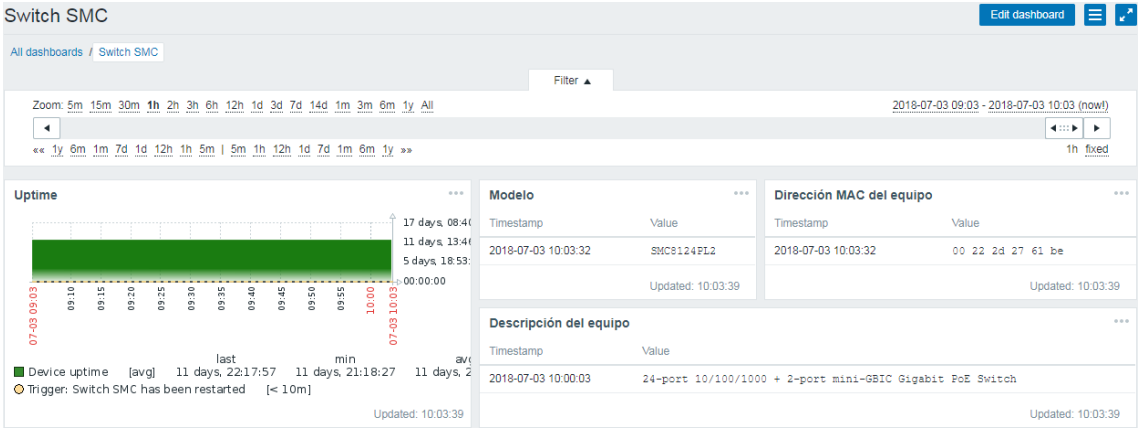


Figura 37. Dashboard switch 1

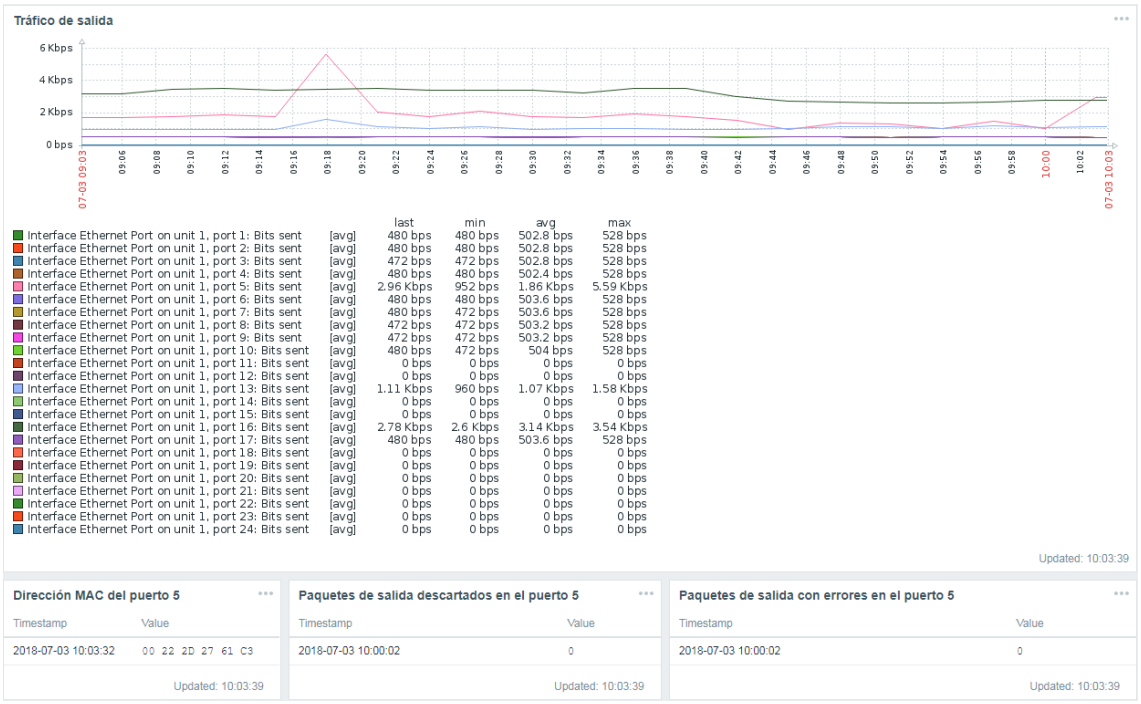


Figura 38. Dashboard switch 2

En ellas se puede ver como se recogen, en una única pantalla, datos como el parámetro "Uptime" o tiempo que lleva activo el equipo desde la última vez que se reinició (en milisegundos), el modelo del equipo que se corresponde con "SMC8124PL2", la dirección MAC y descripción técnica del equipo. Además, se muestra la gráfica anteriormente configurada junto con la dirección MAC y los paquetes de salida descartados y con errores del puerto cinco.

4.2 Router Cisco 2600

Un router es un dispositivo que proporciona conectividad a nivel de red. Su función principal consiste en enviar o encaminar paquetes de datos de una red a otra, es decir, interconectar subredes, entendiendo por subred un conjunto de máquinas IP que se pueden comunicar sin la intervención de un encaminador (mediante puentes de red o un switch), y que por tanto tienen prefijos de red distintos. [24]

El router del laboratorio, modelo Cisco 2600, se encarga de conectar la subred del laboratorio principal con la subred del laboratorio secundario. Este router cuenta con dos interfaces, "FastEthernet 0/0", asociada a la subred del laboratorio principal y, "FastEthernet 0/1", asociada a la subred del laboratorio secundario.

Los ítems seleccionados para la obtención de información acerca del router, aparte de los incluidos en la plantilla "Template Net Network Generic Device SNMPv1", son:

Dirección MAC FastEthernet 0/0:

- Name: Dirección MAC FastEthernet 0/0
- Type: SNMPv1 agent
- Key: dir.macrouter
- SNMP OID: 1.3.6.1.2.1.2.2.1.6.1
- SNMP Community: public
- Port: 161
- Type of information: Character

Dirección MAC FastEthernet 0/1:

- Name: Dirección MAC FastEthernet 0/1
- Type: SNMPv1 agent
- Key: dir.macrouter2
- SNMP OID: 1.3.6.1.2.1.2.2.1.6.3
- SNMP Community: public
- Port: 161
- Type of information: Character

Además de los ya conocidos "UpTime", "Descripción del equipo", "Tráfico de entrada y salida en la interfaz 0/0" y "Tráfico de entrada y salida en la interfaz 0/1".

Seguidamente, se configuran las gráficas "Tráfico de entrada" y "Tráfico de salida", mostradas en las figuras Figura 39 y Figura 40. En ellas se establece una comparación entre la cantidad de tráfico que entra y que sale en las interfaces 0/0 y 0/1.

Como se observa en la Figura 39, la interfaz 0/1 asociada al laboratorio secundario, no recibe apenas tráfico. Esto es debido a que, en el momento de la captura este laboratorio se encontraba apagado o fuera de servicio.

Para recoger estos datos de tráfico, se empleó la herramienta PING, desde un ordenador de la subred del laboratorio principal hacia el router Cisco 2600.

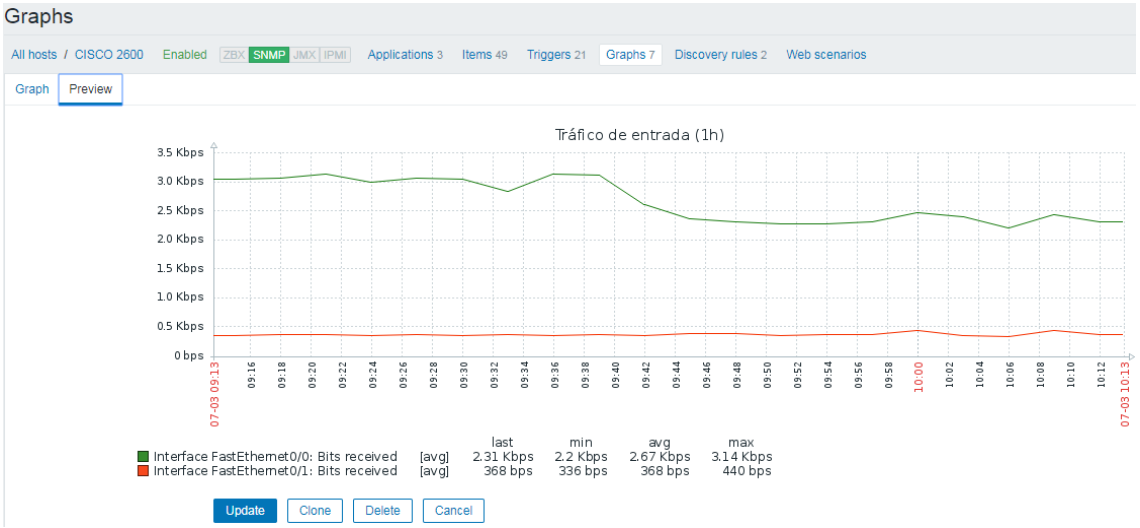


Figura 39. Gráfica router: Tráfico de entrada

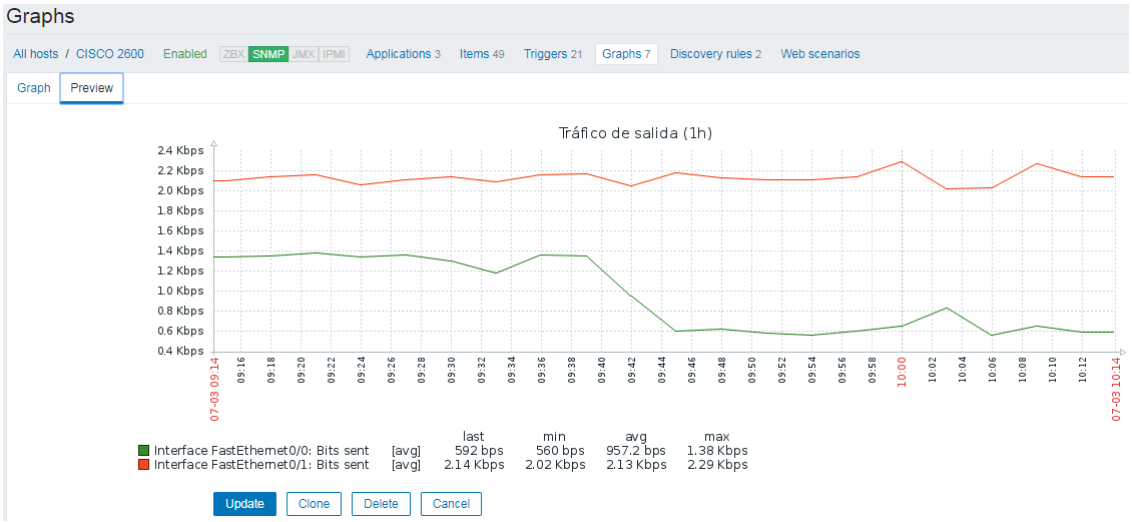


Figura 40. Gráfica router: Tráfico de salida

Para recoger todos estos valores se crea un dashboard como el de la **Error! Reference source not found.**, en el que se muestran el valor del “UpTime”, una descripción detallada del equipo, las direcciones MAC de las interfaces y las dos gráficas anteriormente descritas.

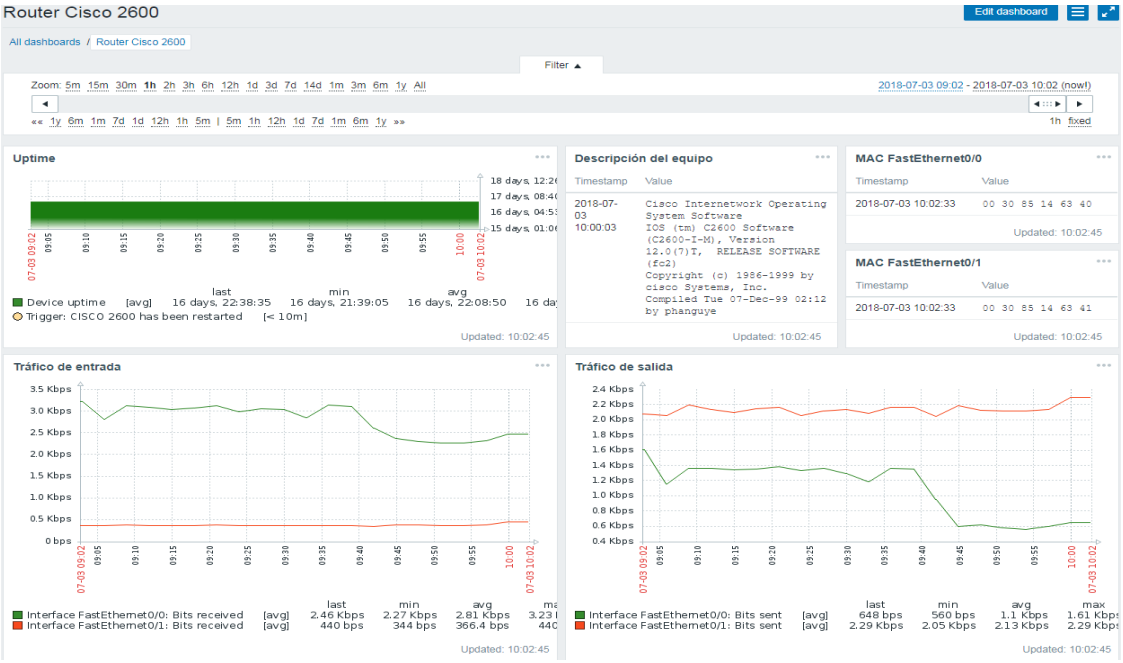


Figura 41. Dashboard router

4.3 Servidor Atlas

El servidor Atlas del laboratorio provee a la red de conexión con el exterior (Internet). Este equipo posee, al igual que el router, dos interfaces. La interfaz “eth0” que comunica la subred del laboratorio con el servidor Atlas y la interfaz “eth1” que proporciona la salida hacia Internet desde el servidor.

Los ítems principales que necesitan ser configurados, a parte de los proporcionados por la plantilla “Template Net Network Generic Device SNMPv1”, son:

Dirección MAC eth0:

- Name: Dirección MAC eth0
- Type: SNMPv1 agent
- Key: dir.maceth0
- SNMP OID: 1.3.6.1.2.1.2.2.1.6.2
- SNMP Community: public
- Port: 161
- Type of information: Character

Dirección MAC eth1:

- Name: Dirección MAC eth1
- Type: SNMPv1 agent
- Key: dir.maceth1
- SNMP OID: 1.3.6.1.2.1.2.2.1.6.3
- SNMP Community: public
- Port: 161
- Type of information: Character

Además, como se hizo con el router Cisco 2600, se crearán dos gráficos, representados en las figuras Figura 42 y Figura 43. En ellas se muestra el tráfico, medido en kilobits por segundo, de entrada y de salida en ambas interfaces.

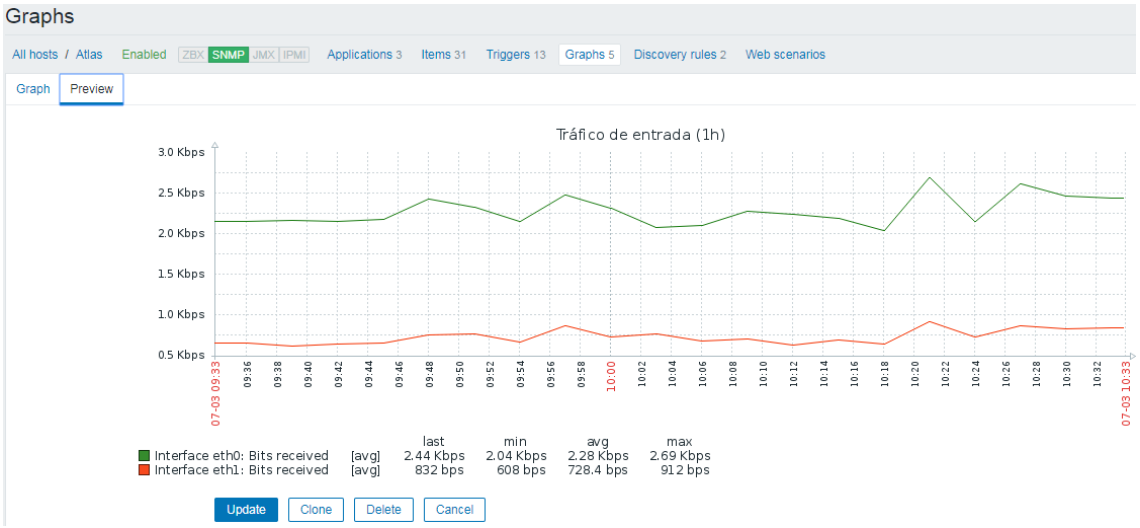


Figura 42. Gráfica Atlas: Tráfico de entrada

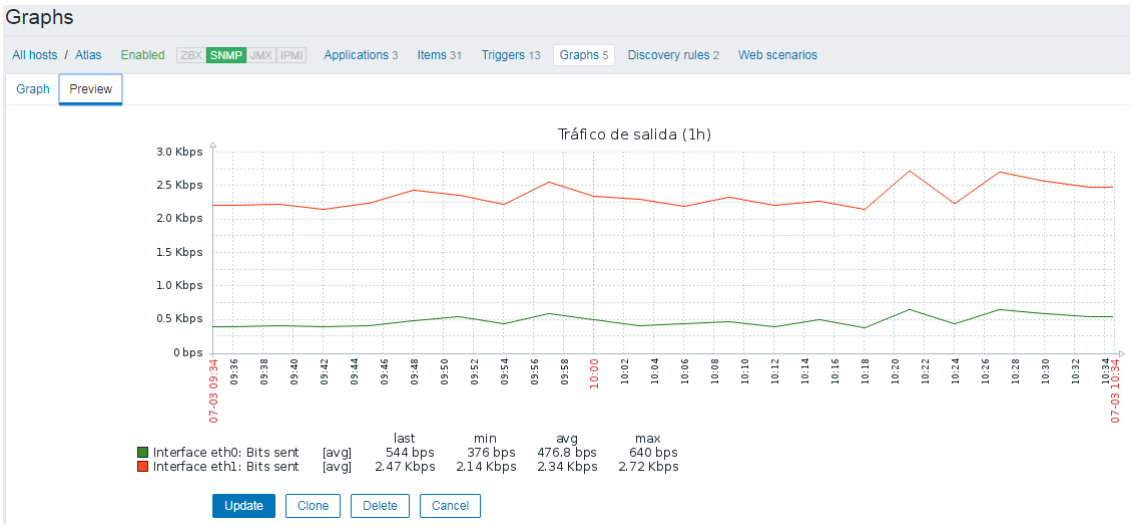


Figura 43. Gráfica Atlas: Tráfico de salida

4.4 Ordenadores Personales

Los tres ordenadores personales del laboratorio que son monitorizados (“Local3”, “Local4” y “Local5”) trabajan originalmente sobre un sistema operativo Windows 7 de 64 bits. Pero se decide implementar en ellos el sistema operativo Ubuntu 16.04 LTS sobre máquinas virtuales creadas con el programa de virtualización “VMWare Player” en cada uno de los PC’s. En este nuevo entorno de trabajo es en el que se realiza la instalación de los agentes nativos de Zabbix, siguiendo los pasos vistos en el apartado “3.3.1 Instalación de agentes”.

En este caso, para la obtención de información acerca de los equipos, ha bastado con los ítems proporcionados por la plantilla “Template OS Linux”. Esta plantilla está configurada para mostrar datos acerca de valores de carga de CPU, espacio en disco del equipo, rendimiento, uso de memoria y tráfico de entrada y salida en equipos que utilicen un sistema operativo tipo Linux.

Para estos tres equipos se ha creado un dashboard como el de la Figura 44. En ella se puede observar cómo se han elegido monitorizar los mismos parámetros en cada equipo para poder, de esta forma, llevar a cabo una comparación entre ellos. Estos parámetros, representados en forma de gráficas, consisten en:

Valores de carga de los procesadores de cada equipo con tres valores para cada uno dependiendo de una media de tiempo desde un minuto (color verde), cinco minutos (color azul) y quince minutos (color rojo). Como se puede observar estos valores resultan pequeños, ya que en el laboratorio no se somete a los equipos a un gran trabajo.

También se representan en el dashboard de la Figura 44 unas gráficas del tráfico de entrada y de salida en cada uno de los equipos, junto con unas gráficas circulares que dan una idea del espacio total disponible en el disco (color rojo) y el espacio libre (color verde), con lo que la diferencia resulta el espacio usado en el disco. Este disco, se refiere al de la máquina virtual utilizada, que reserva un espacio previamente configurado. En este caso de unos 18.58 GB.

Hay que destacar, además, que los valores tan bajos de tráfico observados en el dashboard (alrededor de 1,5 Kbps) se deben a que, en el momento de la captura los ordenadores se encontraban simplemente encendidos.

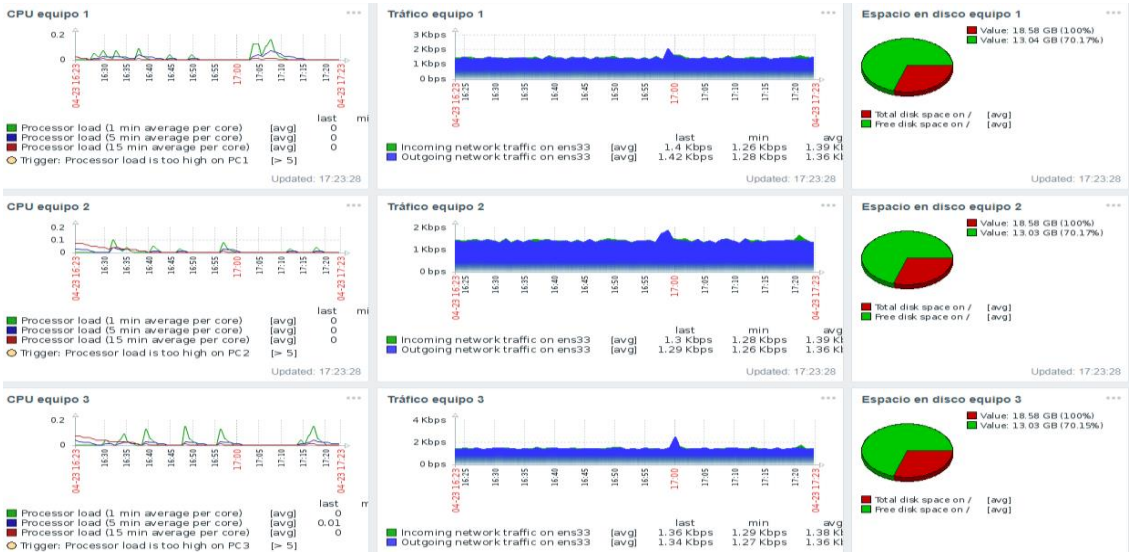


Figura 44. Dashboard equipos

4.5 Mapa de red

En este apartado se expone el mapa, Figura 45, correspondiente a la red del laboratorio creado siguiendo los pasos del punto “2.4.5 Creación de mapas”. En él se puede observar la representación de cada uno de los equipos que conforman la red, así como sus interconexiones.

Cuando los equipos son capaces de comunicarse entre sí, los enlaces se muestran de color verde. Por el contrario, si los enlaces se encuentran caídos, estos tomarán el color rojo para indicar el fallo en la comunicación.

En el mapa de la Figura 45, se muestra la topología de la red, descrita en el apartado “3.2 Descripción de la topología de laboratorio”, de una forma visual e interactiva, ya que se ha configurado de tal manera que, si se quiere obtener información adicional de algún equipo, basta con seleccionar este y, automáticamente la plataforma Zabbix conducirá al usuario hasta él.

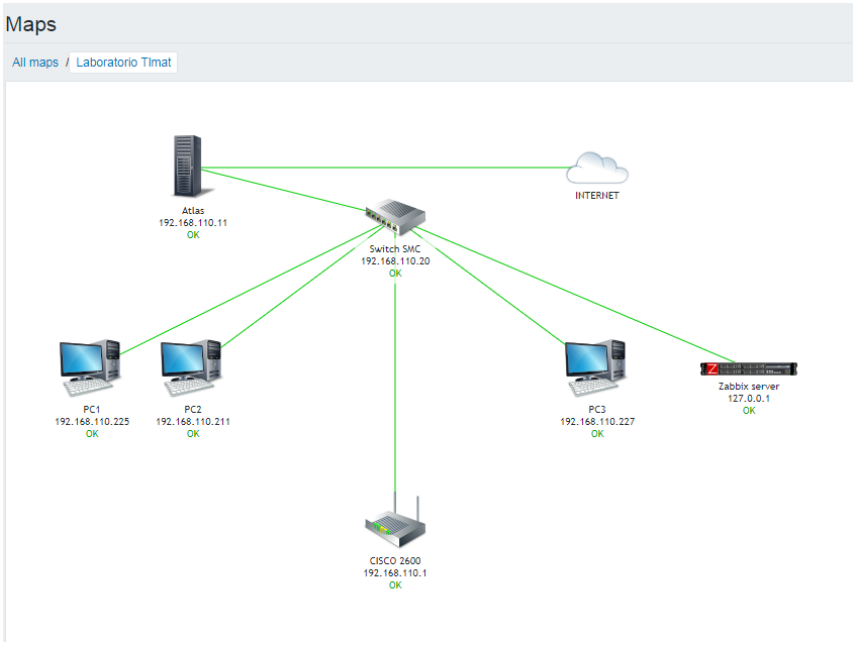


Figura 45. Mapa de la red del laboratorio

Junto con cada equipo se ha configurado que aparezca su dirección IP correspondiente, además de, la visualización del estado de cada uno. En el caso de que el equipo funcione correctamente se mostrará un "OK" de color verde debajo de este. Sin embargo, si surge cualquier incidencia, esta se representará en la pantalla en forma de breve descripción y en color rojo.

Las principales ventajas de utilizar un mapa de red son la capacidad de obtener una visión global de todo el entorno, especialmente útil en entornos con muchos equipos. Y, el averiguar de un simple vistazo que equipos se encuentran apagados o inaccesibles.

Todo esto visto para una implementación pequeña como la del laboratorio podría parecer trivial, pero a medida que el número de equipos a gestionar aumenta, esta tarea se vuelve más complicada.

4.6 Configuración de notificaciones vía e-mail mediante un servidor de correo "PostFix"

Durante la monitorización de la infraestructura las incidencias que van surgiendo son reportadas en la sección "Problemas" del apartado de monitorización, representado en la Figura 46.

Time	Severity	Recovery time	Status	Info	Host	Problem	Duration	Ack	Actions	Tags
10:33:02	Warning	10:34:02	RESOLVED		Switch SMC	High ICMP ping response time	1m	No		
10:30:02	Warning	10:32:02	RESOLVED		Switch SMC	High ICMP ping response time	2m	No		
Today										
2018-05-22 11:55:02	Information		PROBLEM		Switch SMC	Interface Ethernet Port on unit 1, port 9: Ethernet has changed to lower speed than it was before	22h 41m 12s	No		
Yesterday										
2018-05-21 12:04:30	Average		PROBLEM		PC3	Zabbix agent on PC3 is unreachable for 5 minutes	1d 22h 31m	No	Failures 2	
2018-05-21 12:04:30	Average		PROBLEM		PC1	Zabbix agent on PC1 is unreachable for 5 minutes	1d 22h 31m	No		
2018-05-21 12:04:30	Average		PROBLEM		PC2	Zabbix agent on PC2 is unreachable for 5 minutes	1d 22h 31m	No		
2018-05-21 12:00:02	Information		PROBLEM		Switch SMC	Interface Ethernet Port on unit 1, port 4: Ethernet has changed to lower speed than it was before	1d 22h 36m	No		
2018-05-21 12:00:02	Information		PROBLEM		Switch SMC	Interface Ethernet Port on unit 1, port 2: Ethernet has changed to lower speed than it was before	1d 22h 36m	No		

Figura 46. Sección Problemas Zabbix

Como se observa, los problemas surgen a distintas horas por lo que se debe implementar un método de aviso al gestor de la red para que no deba de estar pendiente las 24 horas de esta pantalla en busca de incidencias.

Zabbix provee al gestor de la infraestructura con diferentes soluciones para llevar a cabo esta tarea, como son:

- **E-mail:**

Consiste en el empleo de un servidor de correo para que se pueda producir el envío de incidencias y alertas desde la plataforma Zabbix hacia una dirección de correo electrónico determinada (normalmente la dirección del gestor de la red).

- **SMS:**

Zabbix admite el envío de mensajes SMS utilizando un módem GSM conectado al puerto serie del servidor Zabbix.

- **Jabber:**

Jabber es un protocolo abierto basado en el estándar XML para el intercambio en tiempo real de mensajes y presencia entre dos puntos en Internet.

Al enviar notificaciones, Zabbix intenta buscar primero el registro SRV de Jabber, y si eso falla, utiliza un registro de dirección para ese dominio. Entre los registros SRV de Jabber, se elige el que tenga la prioridad más alta y el peso máximo. Si falla, no se prueban otros registros.

- **Ez Texting:**

Es una solución de mensajería de texto orienta en su mayor parte hacia la empresa.

- **Custom alertscripts:**

Como última solución, existe una alternativa para realizar el envío de alertas creando un "script" personalizado que maneje la notificación a su manera.

Los scripts de alerta se ejecutan en el servidor Zabbix. Estos scripts se encuentran en el directorio definido en el archivo de configuración del servidor, variable: "AlertScriptsPath".

Definitivamente, de entre todas estas soluciones, se decide implementar el envío de notificaciones vía e-mail debido a tener una mayor experiencia en el uso de esta herramienta.

Para configurar el correo electrónico como el canal de entrega de mensajes, se debe configurar el correo electrónico como tipo de medio y asignar direcciones específicas a los usuarios. Para hacerlo, se busca en el menú el botón "Administración", y a la pestaña "Tipos de medios" del submenú. A continuación, se selecciona "Crear tipo de medio" y se rellenan los campos correspondientes como muestra la Figura 47. En ella se puede ver que se especifica un nombre, un correo personal del tipo "SMTP" y un servidor "smtp.gmail.com". En este caso, el correo se corresponde con el gestor de la plataforma Zabbix.

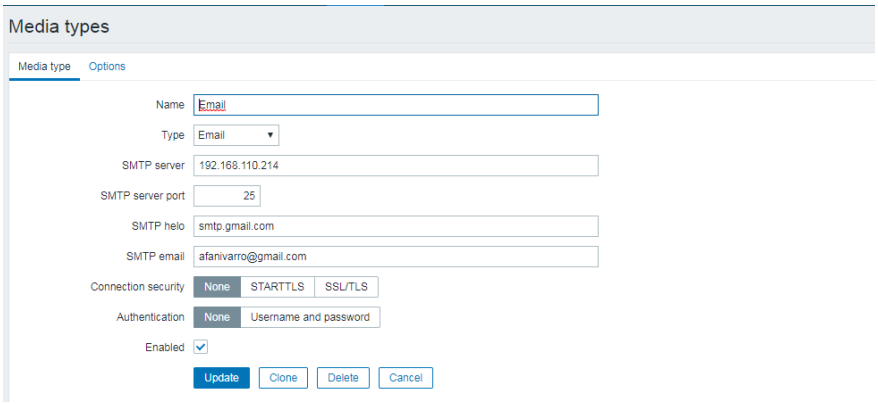


Figura 47. Tipos de medios

Una vez creada la alerta, “Email”, se ha de activar seleccionándola y pulsando en la tecla “Activar” como muestra el proceso de la Figura 48.

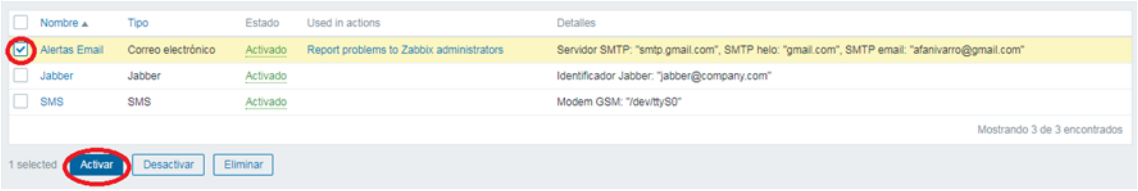


Figura 48. Alertas Email

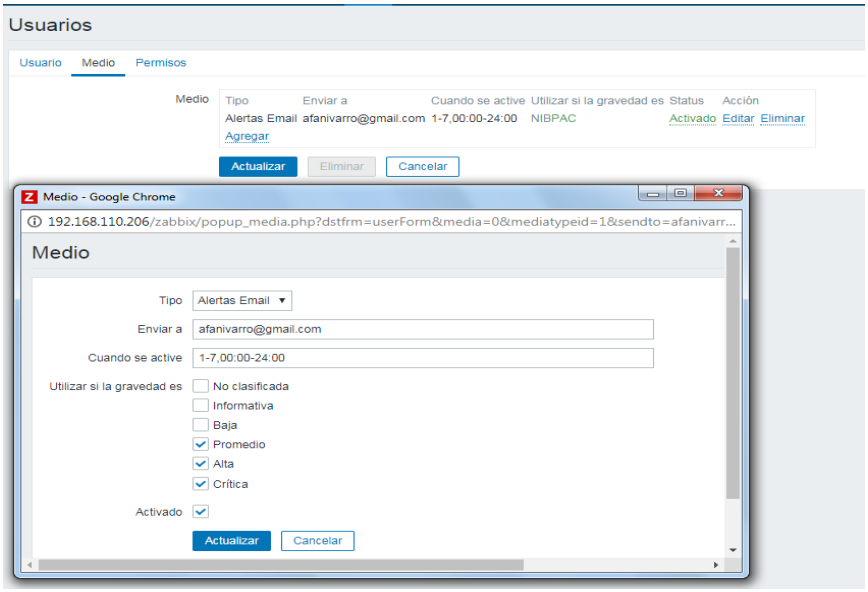


Figura 49. Configuración del medio

En la ventana de configuración del usuario “Admin”, dentro de la pestaña del submenú “Media”, se selecciona “Añadir” para configurar el medio, siguiendo con una configuración como la de la Figura 49. En ella se establecen el tipo de alertas, “Alertas Email”, el destinatario de las alertas, “afanivarro@gmail.com” y, por último, los niveles de gravedad en los cuales la alerta será notificada a través del correo electrónico. Para esta configuración se consideran los niveles “Promedio”, “Alta” y “Crítica”. Todo ello conlleva que sólo se realice el envío de una notificación para cualquiera de esos tres niveles, para cualquier otro, no se produciría ninguna acción.

Finalmente, es necesario realizar la orden del envío de notificaciones al correo electrónico que se ha configurado. Para ello se va al menú “Configuración”, apartado “Acciones”. Allí se crea una acción donde se especifique la opción “enviar mensaje”.

Está operación se encuentra resaltada en negrita en la Figura 50. En ella se pueden observar los campos “Asunto por defecto” y “Mensaje por defecto” en los que se han de definir, en primer lugar, el trigger que defina la incidencia por la cual se lleve a cabo el envío del mensaje. Y, en segundo lugar, la estructura del propio mensaje que el destinatario del correo observará.

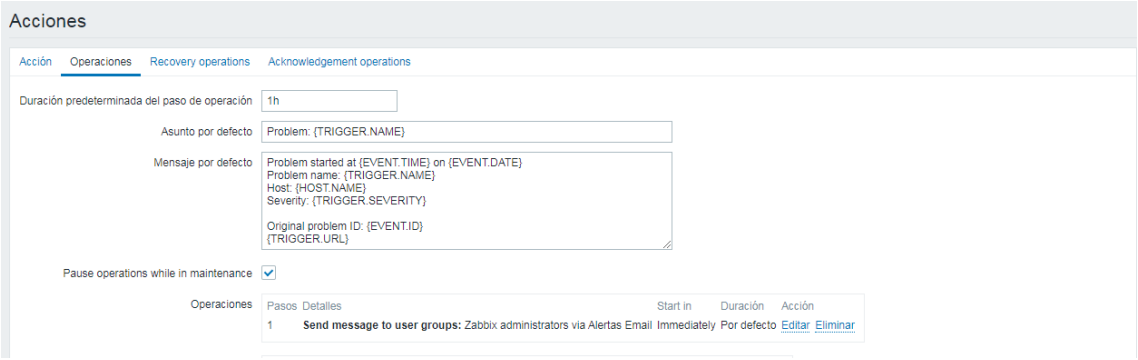


Figura 50. Configuración de acciones

Además, en el campo “Operaciones” se establece la operación que se ha de realizar cuando el valor umbral definido en el trigger es superado. En este caso “Send message to user groups: Zabbix administrators vía Alertas Email”, en el que se especifica, también, el usuario (dentro de la interfaz web de Zabbix) que cuenta con permiso para recibir esa notificación.

Una vez se active el servicio de envío de correos en la plataforma Zabbix, se debe instalar y configurar un servidor de correo. En este caso se ha optado por la instalación de un servidor de correo “PostFix”. Para ello, en el terminal de sistema del servidor Ubuntu se introducen las siguientes líneas:

```
#apt-get install libsasl2-2 libsasl2-modules sasl2-bin mutt postfix
openssl

#nano /etc/default/saslauthd
```

En el fichero “saslauthd” es necesario especificar la contraseña del correo electrónico utilizado (“afanivarro@gmail.com”) para que el servidor PostFix tenga acceso al correo.

En este punto se ha de modificar el parámetro Start=” No” a valor “Yes”

```
#/etc/init.d/saslauthd restart

#cd /etc/postfix

#cp main.cf main.cf.bkp

#nano main.cf
```

Se modifica el archivo “main.cf”, borrando su contenido por completo e introduciendo la siguiente configuración:

```
#SMTP relayhost

relayhost = [smtp.gmail.com]:587

#TLS Settings
```

```
smtp_tls_loglevel = 1

smtp_tls_CAfile = /etc/postfix/certs/CAcert.pem

smtp_tls_cert_file = /etc/postfix/certs/mycert.pem

smtp_tls_key_file = /etc/postfix/certs/mykey.pem

smtp_use_tls = yes

smtpd_tls_CAfile = /etc/postfix/certs/CAcert.pem

smtpd_tls_cert_file = /etc/postfix/certs/mycert.pem

smtpd_tls_key_file = /etc/postfix/certs/mykey.pem

smtpd_tls_received_header = yes

#tls configuration

smtp_sasl_auth_enable = yes

smtp_sasl_password_maps = hash:/etc/postfix/sasl_passwd

smtp_sasl_security_options = noanonymous

smtp_sasl_tls_security_options = noanonymous

#alias de mapeo interno a externo

smtp_generic_maps = hash:/etc/postfix/generic
```

Después se continua con:

```
nano generic

#Conf>

alberto@ubuntu afanivarro@gmail.com

alberto@ubuntu.localdomain afanivarro@gmail.com

nano sasl_passwd

[smtp.gmail.com]:587 afanivarro@gmail.com:" contraseña correo"

mkdir certs

cd certs

openssl dsaparam 1024 -out dsa1024.pem

openssl req -x509 -nodes -days 3650 -newkey dsa:dsa1024.pem -out
mycert.pem -keyout mykey.pem;ln -s mycert.pem CAcert.pem
```

A continuación de este comando se deben rellenar ciertos campos con información personal como nombre del país, localidad, nombre de la organización, FQDN (Fully Qualified Domain Name) y dirección de correo.

```
Openssl req -x509 -new -days 3650 -key /etc/postfix/certs/mykey.pem -
out /etc/postfix/certs/mycert.pem;rm dsa1024.pem

Postmap /etc/postfix/sasl_passwd;postmap /etc/postfix/generic;postmap
/etc/postfix/main.cf

/etc/init.d/postfix restart

echo "E-mail de prueba" | mutt -s "Prueba" afanivarro@gmail.com
```

Este último comando permite realizar una sencilla prueba para comprobar que el servidor de correo actúe correctamente.

Para que el correo electrónico permita el envío de estos correos se debe autorizar el acceso de aplicaciones menos seguras, como aparece reflejado en la Figura 51.

Con esta opción permitida se puede observar el correo de prueba, Figura 52.

```
tail -f /var/log/mail.log
```

Finalmente, se comprueba que el campo status=sent

Después de completar estos pasos se pueden transmitir las incidencias que Zabbix detecte e informar al gestor de la red mediante ellas como se puede apreciar en la Figura 53, donde una alerta por no encontrar el agente activo del equipo tres durante un tiempo de cinco minutos es enviada al correo del gestor.

Figura 51. Permiso de aplicaciones menos seguras

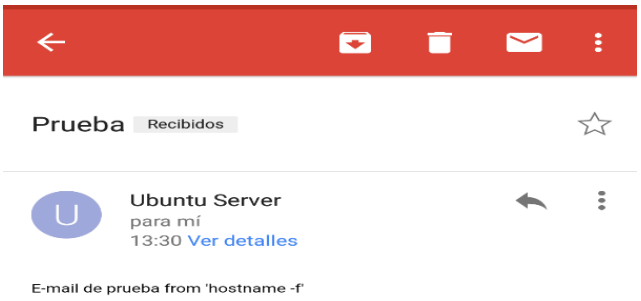


Figura 52. Correo de prueba

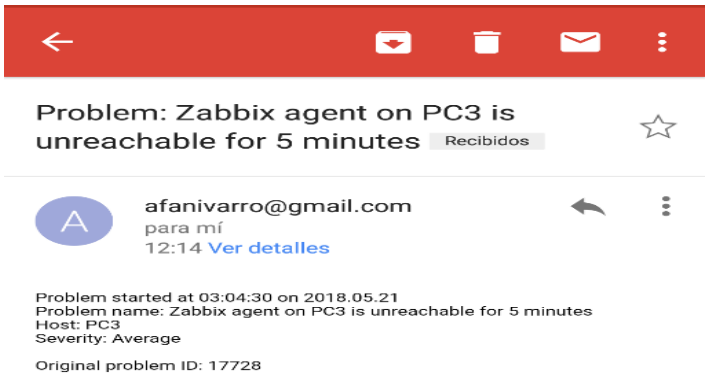


Figura 53. Correo de alerta

CAPÍTULO 5. CONCLUSIONES Y LÍNEAS FUTURAS

Con la realización de este proyecto se ha conseguido implementar un sistema de monitorización de red para el laboratorio docente del grupo de telemática de la Escuela de Ingenieros Industriales y de Telecomunicación de la Universidad de Cantabria.

Para llevar a cabo el desarrollo de este documento se comenzó, en primer lugar, por definir una serie de conceptos teóricos necesarios acerca de la gestión y la monitorización de red, así como la presentación de Zabbix junto con sus características principales.

A continuación, se especifica el proceso a seguir durante la instalación de la plataforma de monitorización, además de realizar una explicación genérica del proceso a seguir en la post-instalación.

Por último, se muestra detalladamente cómo se configura Zabbix para ser capaz de representar la red del laboratorio docente y obtener datos de cada uno de los equipos que la componen.

En cuanto a las conclusiones obtenidas durante la realización de este Trabajo Fin de Grado:

La monitorización de cualquier red siempre resulta rentable si se busca la herramienta adecuada que mejor se adapte a cada tipo de infraestructura. En este caso se trataba de un entorno pequeño de investigación y docencia, por lo que era necesario que la plataforma fuera gratuita y proporcionara información de interés acerca de los equipos componentes de la red. Zabbix cumplió estos requisitos con creces.

Uno de los principales impedimentos fue el hecho de que la documentación oficial no contara con una versión en el idioma castellano. Es por ello, que se utilizó la versión en inglés, aunque las traducciones no siempre se asemejaran a la realidad. También cabe destacar que la mayoría de tutoriales encontrados en la red se impartían en portugués, por lo que el idioma puede ser una desventaja importante para un hispanohablante.

Con respecto a las líneas futuras a implementar:

Sería interesante abordar la instalación y configuración del complemento “Grafana” que permite recoger los datos obtenidos por Zabbix y procesarlos para crear sus propias gráficas y dashboards, que cuentan con un mayor número de detalles y posibilidades.

Además, se podría intentar buscar algún método para generar tráfico en la subred del laboratorio y poder así, asemejarlo a un entorno real. Sin embargo, en caso de que esto no sea posible, se podría optar por llevar a cabo una monitorización durante el transcurso de un período lectivo continuado para, posteriormente analizar los datos obtenidos y proponer mejoras en la red.

También podría ser de utilidad la instalación y comparación en detalle de los distintos sistemas de alertas que Zabbix provee.

Para terminar, sería posible realizar la implementación con otras plataformas existentes en el mercado como, por ejemplo, Pandora FMS.

REFERENCIAS

Para realizar este proyecto se ha consultado información de fuentes distintas como:

[1] Gssi.det.uvigo.es, (2018). [online] Disponible en: http://gssi.det.uvigo.es/users/mramos/public_html/gprsi/gprsi3.pdf [Accedido 4 Jul. 2018].

[2] J. Á. Irastorza Teja, “Introducción a la Gestión de Redes”, [online]. Disponible en: http://www.tlmat.unican.es/siteadmin/submaterials/1551.pdf [Accedido Feb 2018]

[3] Es.wikipedia.org. (2018). Monitorización de redes. [online] Disponible en: https://es.wikipedia.org/wiki/Monitorización_de_redes [Accedido 5 Jul. 2018].

[4] Dinaptica.es. (2018). Monitorización y Gestión de Red | dinaptica. [online] Disponible en: http://www.dinaptica.es/node/18 [Accedido 5 Jul. 2018].

[5] Pablo Turmero, M. (2018). Sistema de gestión de redes y servicios de telecomunicaciones (Presentación PowerPoint) - Monografias.com. [online] Monografias.com. Disponible en: https://www.monografias.com/trabajos102/sistema-gestion-redes-y-servicios-telecomunicaciones/sistema-gestion-redes-y-servicios-telecomunicaciones.shtml [Accedido 5 Jul. 2018].

[6] CCM. (2018). Protocolo SNMP. [online] Disponible en: https://es.ccm.net/contents/280-protocolo-snmp [Accedido 5 Jul. 2018].

[7] Blog.pandorafms.org. (2018). Monitorizacion SNMP: Claves para aprender a usar el protocolo simple de Administración de Red. [online] Disponible en: https://blog.pandorafms.org/es/monitorizacion-snmp/ [Accedido 5 Jul. 2018].

[8] Es.wikipedia.org. (2018). Base de información gestionada. [online] Disponible en: https://es.wikipedia.org/wiki/Base_de_información_gestionada [Accedido 5 Jul. 2018].

[9] Ramonmillan.com. (2018). Gestion de red. [online] Disponible en: https://www.ramonmillan.com/tutoriales/gestionred.php [Accedido 5 Jul. 2018].

[10] Gssi.det.uvigo.es. (2018). [online] Disponible en: http://gssi.det.uvigo.es/users/mramos/public_html/gprsi/gprsi4.pdf [Accedido 5 Jul. 2018].

[11] Es.wikipedia.org. (2018). Nagios. [online] Disponible en: https://es.wikipedia.org/wiki/Nagios [Accedido 5 Jul. 2018].

[12] Martín Pereira Diéguez, Trabajo de fin de grado “Entorno de gestión abierto para un laboratorio de redes de comunicaciones basado en software de monitorización NAGIOS y herramientas SNMP”

[13] Nagios. (2018). Nagios Features - Nagios. [online] Disponible en: https://www.nagios.org/about/features/ [Accedido 5 Jul. 2018].

[14] Es.wikipedia.org. (2018). Pandora FMS. [online] Disponible en: https://es.wikipedia.org/wiki/Pandora_FMS [Accedido 5 Jul. 2018].

[15] Pandora FMS. (2018). Herramientas de monitorización de redes PandoraFMS. [online] Disponible en: https://pandorafms.com/es/herramientas-de-monitorizacion-de-redes/ [Accedido 5 Jul. 2018].

[16] Zabbix.com. (2018). Zabbix features overview. [online] Disponible en: https://www.zabbix.com/product [Accedido 5 Jul. 2018].

[17] Zabbix.com. (2018). Zabbix features overview. [online] Disponible en: https://www.zabbix.com/features [Accedido 5 Jul. 2018].

[18] Network Admin Tools. (2018). Zabbix vs Nagios Comparison for Network and Bandwidth Monitoring. [online] Disponible en: https://www.netadmintools.com/zabbix-vs-nagios-comparison [Accedido 5 Jul. 2018].

[19] Anónimo, (2018). [online] Disponible en: http://www.appperfect.com/blog/agentless-vs-agent-based-monitoring/ [Accedido 5 Jul. 2018].

[20] José M. Ramírez. (2018). *Virtualización de Sistemas Operativos*. [online] Disponible en: https://www.jmramirez.pro/articulo/virtualizacion-de-sistemas-operativos/ [Accedido 5 Jul. 2018].

[21] El Blog de soporteTI. (2018). Networking en VmWare - Modos Bridged y NAT - Vídeo + PowerPoint. [online] Disponible en: https://blog.soporteti.net/networking-en-vmware-modos-bridged-y-nat-video-powerpoint/ [Accedido 5 Jul. 2018].

[22] Digitalocean.com. (2018). ¿Cómo instalar Linux, Apache, MySQL, PHP (LAMP) en Ubuntu 16.04? | DigitalOcean. [online] Disponible en: https://www.digitalocean.com/community/tutorials/como-instalar-linux-apache-mysql-php-lamp-en-ubuntu-16-04-es [Accedido 5 Jul. 2018].

[23] Definición ABC. (2018). Definición de Switch. [online] Disponible en: https://www.definicionabc.com/tecnologia/switch.php [Accedido 5 Jul. 2018].

[24] Es.wikipedia.org. (2018). Router. [online] Disponible en: https://es.wikipedia.org/wiki/Router [Accedido 5 Jul. 2018].