UNIVERSIDAD DE EL SALVADOR

FACULTAD DE INGENIERÍA Y ARQUITECTURA

ESCUELA DE INGENIERÍA ELÉCTRICA

Diseño e implementación de un sistema de gestión remota para equipos de telecomunicaciones.

PRESENTADO POR:

JOSÉ MIGUEL CARRILLO AGUIRRE

PARA OPTAR AL TITULO DE:

INGENIERO ELECTRICISTA

CIUDAD UNIVERSITARIA, OCTUBRE 2015

UNIVERSIDAD DE EL SALVADOR

RECTOR :

ING. MARIO ROBERTO NIETO LOVO

SECRETARIA GENERAL :

DRA. ANA LETICIA ZAVALETA DE AMAYA

FACULTAD DE INGENIERÍA Y ARQUITECTURA

DECANO :

ING. FRANCISCO ANTONIO ALARCÓN SANDOVAL

SECRETARIO :

ING. JULIO ALBERTO PORTILLO

ESCUELA DE INGENIERÍA ELÉCTRICA

DIRECTOR :

MSc. e ING. WILBER CALDERÓN URRUTIA

UNIVERSIDAD DE EL SALVADOR

FACULTAD DE INGENIERÍA Y ARQUITECTURA

ESCUELA DE INGENIERÍA ELÉCTRICA

Trabajo de Graduación previo a la opción al Grado de:

INGENIERO ELECTRICISTA

Título :

Diseño e implementación de un sistema de gestión remota para equipos de telecomunicaciones.

Presentado por :

JOSÉ MIGUEL CARRILLO AGUIRRE

Trabajo de Graduación Aprobado por:

Docente Asesor :

ING. WERNER DAVID MELÉNDEZ VALLE

San Salvador, Octubre 2015

Trabajo de Graduación Aprobado por:

Docente Asesor :

ING. WERNER DAVID MELÉNDEZ VALLE

ACTA DE CONSTANCIA DE NOTA Y DEFENSA FINAL

En esta fecha, jueves 27 de agosto de 2015, en la Sala de Reuniones de la Escuela de Ingeniería Eléctrica, a las 4:00 horas, en presencia de las siguientes autoridades de la Escuela de Ingeniería Eléctrica de la Universidad de El Salvador:

1. MSc. e Ing. José Wilber Calderón Urrutia
   Director

   Firma: Wilber Calderón

2. MSc. e Ing. Salvador de Jesús Germán
   Secretario

   Firma:

ESCUELA DE INGENIERIA ELECTRICA DIRECCION

Y, con el Honorable Jurado de Evaluación integrado por las personas siguientes:

Firma:

1- Ing. Werner David Melendez Valle

2- Ing. Armando Martínez Calderón

3- MSc. Carlos Osmín Pocasangre Jiménez

Se efectuó la defensa final reglamentaria del Trabajo de Graduación:

Diseño e implementación de un sistema de gestión remota para equipos de telecomunicaciones.

A cargo del Bachiller:

- Carrillo Aguirre José Miguel

Habiendo obtenido en el presente Trabajo una nota promedio de la defensa final: 8.0

(OCHO PUNTO CERO)

AGRADECIMIENTOS

Agradezco a Dios todo poderoso por permitirme cumplir este logro que culmina este día, doy las gracias a mi padre, Miguel Carrillo, por ser el mayor de mis apoyos y un modelo siempre a seguir y a mi madre, Elena Aguirre por siempre creer en mí y motivarme a seguir adelante en cada momento. Agradezco a mi hermano Jonathan por ser mi compañero y amigo en todo este proceso. A mis abuelas Carmen y Mercedes por sus infinitas oraciones y por estar pendiente todo este tiempo. A Denisse por acompañarme en la mayoría de este viaje que este día culmina. Finalmente, Al Ing. Werner Melendez por su apoyo, guía y paciencia en el desarrollo de este proyecto, a todos ustedes les debo este triunfo y agradezco infinitamente.

ÍNDICE

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

OBJETIVOS

GENERALES.

Examinar y proponer, diferentes opciones de monitoreo y gestión de bajo costo, que se puedan utilizar en sistemas de comunicaciones.

ESPECÍFICOS.

- Analizar alternativas de software de gestión de bajo costo, disponibles en el mercado que puedan utilizarse para el monitoreo de sistemas de comunicaciones y demás equipos asociados (tipo generadores, Aires Acondicionados, etc.)

- Implementar las interfaces que sean necesarias para integrar (al sistema de monitoreo) los equipos auxiliares que no dispongan de capacidad SNMP.

- Proponer alternativas para incorporar al sistema de monitoreo, facilidades de gestión o ejecución de software de terceros

ALCANCES

Estudiar las capacidades de distintos tipos de software de monitoreo y gestión disponibles en el mercado, y analizar su viabilidad de aplicación en un prototipo de sistema de comunicación; el cual deberá incluir equipos de radio, de acceso (tipo routers y switches), equipos auxiliares y demás afines. Además, el sistema a proponer, deberá permitir ejecutar software de gestión de terceros, y facilidades para incorporar nuevos elementos conforme sea necesario. Los parámetros a monitorear, deberán centrarse en aquellos que afecten la disponibilidad y calidad del servicio.

CAPITULO I

1. ALTERNATIVAS DE MONITOREO REMOTO DE SISTEMAS DE TELECOMUNICACIONES

1.1 INTRODUCCIÓN A LOS SISTEMAS DE MONITOREO

El término Monitoreo de Sistemas de Telecomunicaciones describe el uso de un medio que verifica el estado de los diversos módulos que conforman dicho sistema, en busca de componentes defectuosos o con mal funcionamiento, para luego notificar dicha situación a los administradores de la red, mediante diversas modalidades.

Los equipos que se incluyen en redes de comunicaciones en la actualidad son muy diversos, es común ver equipos muy amigables al usuario y de fácil gestión, intuitivos y que tienen la facilidad de ser monitoreados de manera nativa, equipos recientes como Routers, Switchs, o servidores entre otros. También es común encontrar equipos que no tienen prestaciones tan avanzadas y su gestión es muy limitada o nula, tal es el caso de dispositivos como switchs no programables, equipos de red desfasados, etc. Existen además elementos auxiliares (en especial aquellos fabricados hace tres años o más) que por su naturaleza no están diseñados para incorporar algún tipo de monitoreo o control, por ejemplo, plantas eléctricas, bancos de baterías, aires acondicionados, UPS´s, etc.

En general los sistemas de monitoreo permiten a los administradores conocer en tiempo real el estado de los dispositivos que conforman un sistema de telecomunicaciones, lo que les permite tomar acciones manuales o automáticas para garantizar el óptimo funcionamiento de las redes, por ejemplo, cambiar el enlace de datos entre un punto y otro para re dirigir el tráfico o dejar fuera de la red a un equipo que está causando conflictos mayores en la red completa, son acciones que se pueden tomar a partir del eficiente monitoreo de las redes.

Por ejemplo, para determinar el estatus de un servidor web, software de monitoreo que puede enviar, periódicamente, peticiones HTTP (Protocolo de Transferencia de Hipertexto) para obtener páginas; para un servidor de correo electrónico, enviar mensajes mediante SMTP (Protocolo de Transferencia de Correo Simple), para luego ser retirados mediante IMAP (Protocolo de Acceso a Mensajes de Internet) o POP3 (Protocolo Post Office).

Comúnmente, los datos evaluados son tiempo de respuesta y disponibilidad (o uptime), aunque estadísticas tales como consistencia y fiabilidad han ganado popularidad. La generalizada instalación de dispositivos de optimización para redes de área extensa tiene un efecto adverso en la mayoría del software de monitoreo, especialmente al intentar medir el tiempo de respuesta de punto a punto de manera precisa, dado el límite visibilidad de ida y vuelta.

Las fallas de peticiones de estado, tales como que la conexión no pudo ser establecida, el tiempo de espera agotado, entre otros, usualmente produce una acción desde del sistema de monitoreo. Estas acciones pueden variar: una alarma puede ser enviada al administrador, ejecución automática de mecanismos de controles de fallas, etcétera.

Este capítulo tiene como finalidad la introducción de los sistemas de monitoreo y gestión, presentando los conceptos básicos, tipos de monitoreo, protocolos y la información de los equipos que componen las redes de monitoreo.

1.2 TIPOS DE MONITOREO

Existe dos clasificaciones de monitoreo, basadas en la influencia en el tráfico de la red a causa de las mediciones y la información que buscan obtener de ella, bajo estos criterios podemos clasificar las acciones de monitoreo como activo o pasivo.

1.2.1 MONITOREO ACTIVO:

Este tipo de monitoreo se realiza introduciendo paquetes de pruebas en la red, o enviando paquetes a determinadas aplicaciones y midiendo sus tiempos de respuesta. Este enfoque tiene la característica de agregar tráfico en la red y es empleado para medir el rendimiento de la misma.

Basado en ICMP

- Diagnosticar problemas en la red.
- Detectar retardo, pérdida de paquetes.
- RTT
- Disponibilidad de host y redes.

Basado en TCP

- Tasa de transferencia.
- Diagnosticar problemas a nivel de aplicación

Basado en UDP

- Pérdida de paquetes en un sentido (one – way)
- RTT (tracerroute)

1.2.2 MONITOREO PASIVO:

Este enfoque se basa en la obtención de datos a partir de recolectar y analizar el tráfico que circula por la red. Se emplean diversos dispositivos como sniffers, routers, computadoras con software de análisis de tráfico y en general dispositivos con soporte para SNMP, RMON y Netflow. Este enfoque no agrega tráfico a la red como lo hace el activo y es utilizado para caracterizar el tráfico en la red y para contabilizar su uso, algunos ejemplos de este tipo son:

Mediante SNMP:

Esta técnica es utilizada para obtener estadísticas sobre la utilización de ancho de banda en los dispositivos de red, estado de los dispositivos o comportamiento dentro de la red, para ello se requiere tener acceso a dichos dispositivos. Al mismo tiempo, este protocolo genera paquetes llamados traps que indican que un evento inusual se ha producido. Esta técnica será retomada a mayor detalle más adelante en este capitulo

1.2.3 OTROS MÉTODOS DE ACCESO:

Se pueden realizar scripts que tengan acceso a dispositivos remotos para obtener información importante a monitorear.

Captura de tráfico:

Se utiliza para caracterizar el tráfico de red, es decir, para identificar el tipo de aplicaciones que son más utilizadas. Se puede implementar haciendo uso de dispositivos probe que envíen información mediante RMON o a través de un dispositivo intermedio con una aplicación capaz de clasificar el tráfico por aplicación, direcciones IP origen y destino, puertos origen y destino, etc.

Flujos:

También utilizado para identificar el tipo de tráfico utilizado en la red. Un flujo es un conjunto de paquetes con:

1. La misma dirección
2. El mismo puerto TCP origen y destino
3. El mismo tipo de aplicación.

4. Los flujos pueden ser obtenidos de routers o mediante dispositivos que sean capaces de capturar tráfico y transformarlo en flujos. También es usado para tareas de facturación.

Las alarmas son consideradas como eventos con comportamiento inusual. Las alarmas más comunes son las que reportan cuando el estado operacional de un dispositivo o servicio cambia. Existen otros tipos de alarmas basado en patrones previamente definidos en métricas, son valores máximos conocidos como umbrales o threshold. Cuando estos patrones son superados se produce una alarma, ya que es considerado como un comportamiento fuera del patrón. Algunos tipos de alarmas son:

- Alarmas de procesamiento.
- Alarmas de conectividad.
- Alarmas ambientales.
- Alarmas de utilización.
- Alarmas de disponibilidad.

1.3 PROTOCOLO SNMP

El SNMP es un protocolo de la capa de aplicación que facilita el intercambio de información de gestión entre dispositivos de una red. SNMP es parte de TCP/IP. SNMP permite a los administradores de red supervisar el rendimiento de la red, buscar y resolver sus problemas y planear el crecimiento de la red.

Dos versiones de SNMP existen: SNMP V1 y SNMP V2, ambas versiones tienen un numero de características en común, pero SNMP V2 ofrece mejoras en las operaciones del protocolo; otra versión de SNMP V3 ofrece mejoras sobre los aspectos de seguridad pero aún está en revisión.

Una red administrada con SNMP consiste de tres componentes fundamentales:

- Dispositivos administrados. Managed Devices (MD)
- Agentes Agent
- Sistemas administrados de Red. NMS

Un Dispositivo administrado (MD), es un nodo de red que contiene un agente SNMP y que reside en una red administrada. Los Dispositivos administrados colectan y almacenan información y hacen que esta información esté disponible al NMS´s utilizando SNMP. Los Managed Devices, algunas veces llamados elementos de red, pueden ser routers y servidores de acceso, switches y bridges, hubs, computadoras anfitrionas o impresoras.

Un agente es un módulo de Software de gestión de red que reside en un Managed Device. Un Agente tiene conocimiento local de información (sobre su memoria,

número de paquetes recibidos enviados, direcciones IP, rutas, etc.) y traduce esa información en una forma de formato compatible con SNMP.

Un NMS ejecuta aplicaciones que monitorean y controlan Managed Devices. Los NMS´s proporcionan la mayor parte de recursos de procesamiento y memoria requeridos para la gestión de la red. Uno o más NMS´s deben existir en cualquier red administrada.

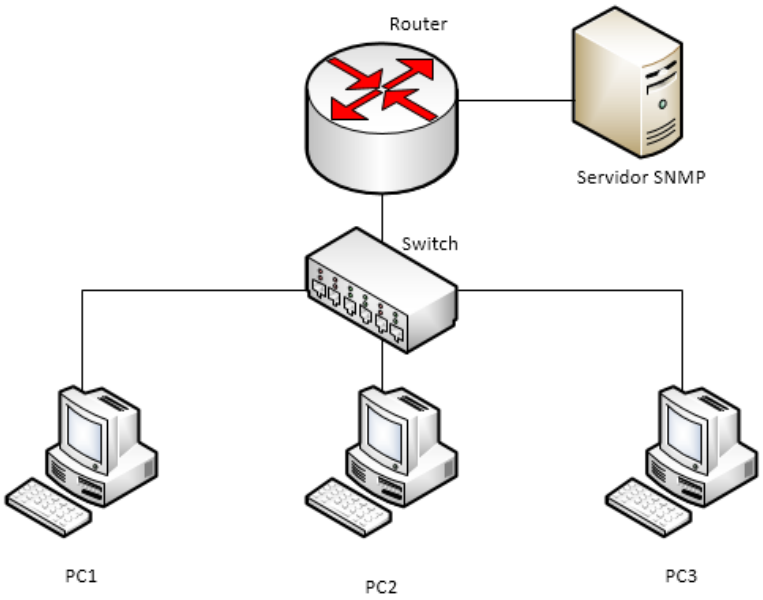


Figura 1: Ejemplo de red básica

Se puede ejemplificar una red básica SNMP a partir de la figura 1, para el caso un switch y un Router representan dispositivos administrados o MD, además de las computadoras que cuenten con un Agente SNMP, es decir software dedicado a recopilar información importante de sus anfitriones y procesarlo para ser transmitido en dicho protocolo, finalmente el servidor que contiene un software de monitoreo o gestión es considerada una estación de monitoreo de red o NMS.

1.3.1 COMANDOS BÁSICOS SNMP

Los Managed Devices son supervisados y controlados utilizando cuatro comandos SNMP básicos:

- Read
- Write
- Trap
- Operaciones de recorrido (Traversal Operations).

El comando READ es utilizado por un NMS para supervisar los Managed Devices. El NMS examina diferentes variables que son mantenidas por los MD.

El comando WRITE es utilizado por un NMS para controlar los MD. El NMS cambia los valores de las variables almacenadas dentro de los Managed Devices

El comando TRAP es utilizado por los Managed Devices para reportar eventos de forma asíncrona a los Network Management Systems (NMS). Cuando cierto tipo de eventos ocurren, un MD envía un TRAP hacia el NMS.

Las operaciones de recorrido o Traversal Operations son utilizadas por los NMS para determinar cuáles variables son soportadas por los MD y obtener secuencialmente información en una tabla de variables, tal como una tabla de enrutamiento.

1.3.2 BASE DE INFORMACIÓN DE ADMINISTRACIÓN (MANAGEMENT INFORMATION BASE, MIB)

Una Base de Información de Administración (Management Information Base, MIB) es una colección de información que está organizada jerárquicamente. Las MIB's son accedidas usando un protocolo de administración de red, como por ejemplo, SNMP.

Un objeto administrado (algunas veces llamado objeto MIB, objeto, o MIB) es uno de cualquier número de características específicas de un dispositivo administrado. Los objetos administrados están compuestos de una o más instancias de objeto, que son esencialmente variables.

Existen dos tipos de objetos administrados: Escalares y tabulares. Los objetos escalares definen una simple instancia de objeto. Los objetos tabulares definen múltiples instancias de objeto relacionadas que están agrupadas conjuntamente en tablas MIB.

Un ejemplo de un objeto administrado es atInput, que es un objeto escalar que contiene una simple instancia de objeto, el valor entero que indica el número total de paquetes AppleTalk de entrada sobre una interfaz de un router.

Un identificador de objeto (object ID) identifica únicamente a un objeto administrado en la jerarquía MIB. La jerarquía MIB puede ser representada como un árbol con una raíz anónima y los niveles, que son asignados por diferentes organizaciones.

El árbol MIB ilustra las variadas jerarquías asignadas por las diferentes organizaciones

Los identificadores de los objetos ubicados en la parte superior del árbol pertenecen a diferentes organizaciones estándares, mientras los identificadores de los objetos ubicados en la parte inferior del árbol son colocados por las organizaciones asociadas.

Los vendedores pueden definir ramas privadas que incluyen los objetos administrados para sus propios productos. Las MIB's que no han sido estandarizadas típicamente están localizadas en la rama experimental.

El objeto administrado atInput podría ser identificado por el nombre de objeto **iso.identified-organization.dod.internet.private.enterprise.cisco.temporary.AppleTalk.atInput** o por el descriptor de objeto equivalente 1.3.6.1.4.1.9.3.3.1.

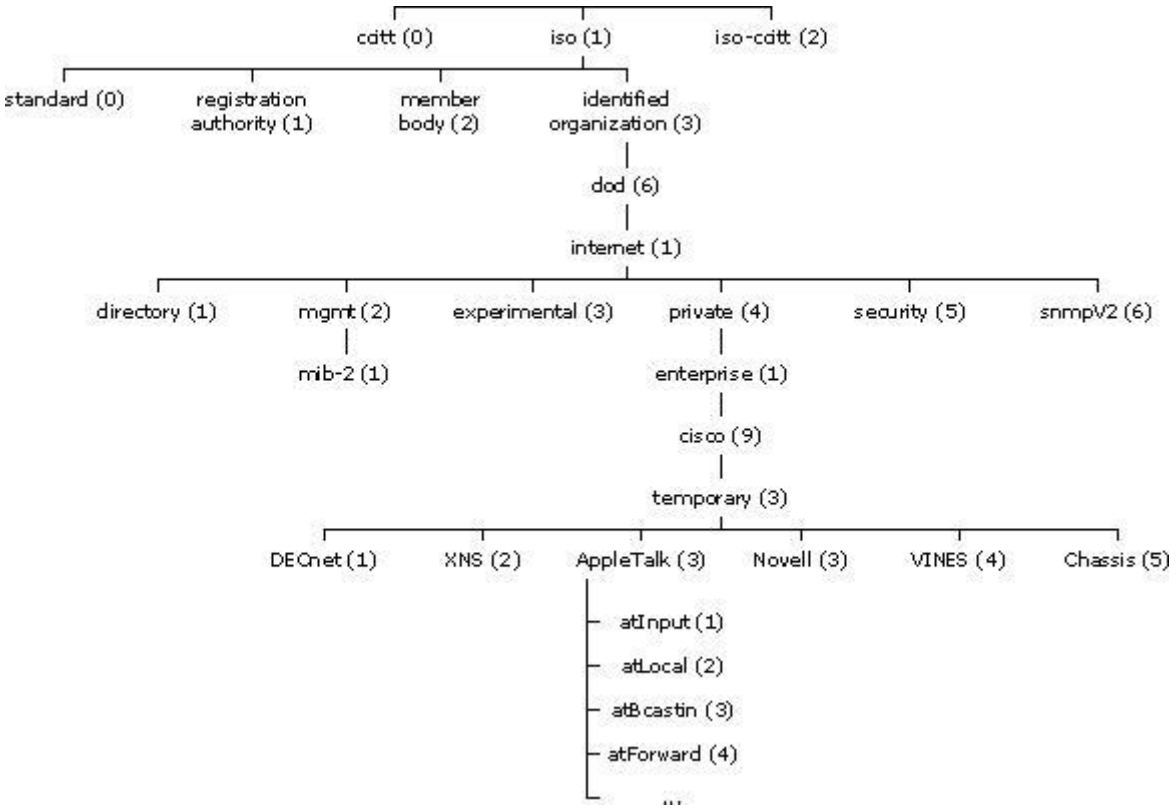


Figura 2: Ejemplo de Árbol MIB para la lectura de paquetes AppleTalk en la interfaz de un router

El corazón del árbol MIB se encuentra compuesto de varios grupos de objetos, los cuales en su conjunto son llamados mib-2. Los grupos son los siguientes:

- System (1)
- Interfaces (2)
- AT (3)
- IP (4)
- ICMP (5)
- TCP (6)
- UDP (7)
- EGP (8)
- Transmission (10)
- SNMP (11)

1.3.3 DETALLES DEL PROTOCOLO

SNMP opera en la capa de aplicación del conjunto de protocolos de Internet (capa siete del modelo OSI). El agente SNMP recibe solicitudes en el puerto UDP 161. El administrador puede enviar solicitudes de cualquier puerto de origen disponible para el puerto 161 en el agente. La respuesta del agente será enviado de vuelta al puerto de origen en el gestor. El administrador recibe notificaciones (Tramps e InformRequests) en el puerto 162. El agente puede generar notificaciones desde cualquier puerto disponible. Cuando se utiliza con Transport Layer Security o datagramas de Transport Layer Security solicitudes se reciben en el puerto 10161 y tramps se envían al puerto 10162. SNMPv1 especifica cinco centrales unidades de datos de protocolo (PDU). Otros dos PDU, GetBulkRequest e InformRequest se añadieron en SNMPv2 y prorrogados a SNMPv3.

Todas las PDU SNMP se construyen de la siguiente manera:

- Cabecera IP
- Encabezado UDP versión comunidad
- Tipo de PDU
- Petición-ID
- Error de estado
- Índice de errores
- Enlaces de variables

1.3.4 MENSAJES SNMP

Para realizar las operaciones básicas de administración anteriormente nombradas, el protocolo SNMP utiliza un servicio no orientado a la conexión (UDP) para enviar un pequeño grupo de mensajes (PDUs) entre los administradores y agentes. La utilización de un mecanismo de este tipo asegura que las tareas de administración de red no afectarán al rendimiento global de la misma, ya que se evita la utilización de mecanismos de control y recuperación como los de un servicio orientado a la conexión, por ejemplo TCP.

Los puertos comúnmente utilizados para SNMP son: 161 SNMP y 162 para SNMP Trap.

Los paquetes utilizados para enviar consultas y respuestas SNMP poseen el siguiente formato:

Versión	Comunidad	SNMP PDU

Figura 3: Formato de mensaje SNMP

Versión: Número de versión de protocolo que se está utilizando (por ejemplo 1 para SNMPv1);

Comunidad: Nombre o palabra clave que se usa para la autenticación. Generalmente existe una comunidad de lectura llamada "public" y una comunidad de escritura llamada "private";

SNMP PDU: Contenido de la unidad de datos del protocolo, el que depende de la operación que se ejecute.

Los mensajes GetRequest, GetNextRequest, SetRequest y GetResponse utilizan la siguiente estructura en el campo SNMP PDU:

Tipo	Identificador	Estado de error	Índice de error	Enlazado de variables

Figura 4: Estructura PDU SNMP

Identificador: Es un número utilizado por el NMS y el agente para enviar solicitudes y respuesta diferentes en forma simultánea;

Estado e índice de error: Sólo se usan en los mensajes GetResponse (en las consultas siempre se utiliza cero). El campo "índice de error" sólo se usa cuando "estado de error" es distinto de 0 y posee el objetivo de proporcionar información adicional sobre la causa del problema. El campo "estado de error" puede tener los siguientes valores:

- 0: No hay error
- 1: Demasiado grande

- 2: No existe esa variable
- 3: Valor incorrecto
- 4: El valor es de solo lectura
- 5: Error genérico

Enlazado de variables: Es una serie de nombres de variables con sus valores correspondientes (codificados en ASN.1).

GetRequest: A través de este mensaje el NMS solicita al agente retornar el valor de un objeto de interés mediante su nombre. En respuesta el agente envía una respuesta indicando el éxito o fracaso de la petición. Si la petición fue correcta, el mensaje resultante también contendrá el valor del objeto solicitado. Este mensaje puede ser usado para recoger un valor de un objeto, o varios valores de varios objetos, a través del uso de listas.

GetNextRequest: Este mensaje es usado para recorrer una tabla de objetos. Una vez que se ha usado un mensaje GetRequest para recoger el valor de un objeto, puede ser utilizado el mensaje GetNextRequest para repetir la operación con el siguiente objeto de la tabla. Siempre el resultado de la operación anterior será utilizado para la nueva consulta. De esta forma un NMS puede recorrer una tabla de longitud variable hasta que haya extraído toda la información para cada fila existente.

SetRequest: Este tipo de mensaje es utilizado por el NMS para solicitar a un agente modificar valores de objetos. Para realizar esta operación el NMS envía al agente una lista de nombres de objetos con sus correspondientes valores.

GetResponse: Este mensaje es usado por el agente para responder un mensaje GetRequest, GetNextRequest, o SetRequest. En el campo "Identificador de Request" lleva el mismo identificador que el "request" al que está respondiendo.

Trap: Una trap es generada por el agente para reportar ciertas condiciones y cambios de estado a un proceso de administración. El formato de la PDU es diferente:

Tipo	Enterprise	Dirección del agente	Tipo genérico de trap	Tipo específico de trap	Timestamp	Enlazado de variables

Figura 5: PDU SNMP para TRAP

- **Enterprise:** Identificación del subsistema de gestión que ha emitido el trap;
- **Dirección del agente:** Dirección IP del agente que ha emitido el trap;
- **Tipo genérico de trap:**
  - Cold start (0): Indica que el agente ha sido inicializado o reinicializado;
  - Warm start (1): Indica que la configuración del agente ha cambiado;
  - Link down (2): Indica que una interfaz de comunicación se encuentra fuera de servicio (inactiva);
  - Link up (3): Indica que una interfaz de comunicación se encuentra en servicio (activa);
  - Authentication failure (4): Indica que el agente ha recibido un requerimiento de un NMS no autorizado (normalmente controlado por una comunidad);
  - EGP neighbor loss (5): Indica que en sistemas en que los routers están utilizando el protocolo EGP, un equipo colindante se encuentra fuera de servicio;
  - Enterprise (6): En esta categoría se encuentran todos los nuevos traps incluidos por los vendedores.
  - Tipo específico de trap: Es usado para traps privados (de fabricantes), así como para precisar la información de un determinado trap genérico;
  - Timestamp: Indica el tiempo que ha transcurrido entre la reinicialización del agente y la generación del trap;
  - Enlazado de variables: Se utiliza para proporcionar información adicional sobre la causa del mensaje.

GetBulkRequest: Este mensaje es usado por un NMS que utiliza la versión 2 ó 3 del protocolo SNMP típicamente cuando es requerida una larga transmisión de datos, tal como la recuperación de largas tablas. En este sentido es similar al mensaje GetNextRequest usado en la versión 1 del protocolo, sin embargo, GetBulkRequest es un mensaje que implica un método mucho más rápido y eficiente, ya que a través de un solo mensaje es posible solicitar la totalidad de la tabla.

InformRequest: Un NMS que utiliza la versión 2 ó 3 del protocolo SNMP transmite un mensaje de este tipo a otro NMS con las mismas características, para notificar información sobre objetos administrados, utilizando el protocolo de nivel 4(osi) TCP, y enviara el InformRequest hasta que tenga un acuse de recibo.

1.4 ALTERNATIVAS SOFTWARE PARA MONITOREO

En el presente trabajo se analizan diversas opciones basadas en el protocolo de monitoreo SNMP, de la gran mayoría de opciones existentes en el mercado se han tomado como criterios de selección que el software utilizado presente opciones de licencia Open Source, facilidad de lectura de MiB creadas para dispositivos que no poseen nativamente propiedades del protocolo SNMP y la disponibilidad de material extenso de referencia para uso de dicho software.

1.4.1 CACTI

Es una completa solución para la generación de gráficos en red, diseñada para aprovechar el poder de almacenamiento y la funcionalidad para gráficas que poseen las aplicaciones RRDtool. Esta herramienta, desarrollada en PHP, provee un pooler ágil, plantillas de gráficos avanzadas, múltiples métodos para la recopilación de datos, y manejo de usuarios. Tiene una interfaz de usuario fácil de usar, que resulta conveniente para instalaciones del tamaño de una LAN, así como también para redes complejas con cientos de dispositivos.

RRDtool es el acrónimo de Round Robin Database tool, o sea que se trata de una herramienta que trabaja con una BD que maneja Planificación Round-robin. Esta técnica trabaja con una cantidad fija de datos y un puntero al elemento actual. El modo en que trabaja una base de datos utilizando Round Robin es el siguiente; se trata la BD como si fuera un círculo, sobrescribiendo los datos almacenados, una vez alcanzada la capacidad de la BD. La capacidad de la BD depende de la cantidad de información como historial que se quiera conservar.

Cacti representa gráficamente los datos almacenados en la RRD, pudiendo ser estos cualquier valor censado, voltaje, corriente, temperatura, ancho de banda usado, etc. La ventaja de la RRDTool radica poder alimentarla con los datos obtenidos mediante cualquier sensor y la herramienta crea una base de datos, almacena los datos en ella, recupera estos datos y basándose en ellos CACTI crea gráficos en formatos PNG

Para manejar la recopilación de datos, se le puede pasar a Cacti la ruta a cualquier script o comando junto con cualquier dato que el usuario necesitare ingresar; Cacti reunirá estos datos, introduciendo este trabajo en el cron (para el caso de un sistema operativo Linux) y cargará los datos en la BD MySQL y los archivos de Planificación Round-robin que deba actualizar.

Una fuente de datos también puede ser creada. Por ejemplo, si se quisiera representar en una gráfica los tiempos de ping de un host, se podría crear una fuente

de datos, utilizando un script que haga ping a un host y devuelva el valor en milisegundos. Después de definir opciones para la RRDtool, como ser la forma de almacenar los datos, uno puede definir cualquier información adicional que la fuente de entrada de datos requiera, como ser en este caso, la IP del host al cual hacer el ping. Luego que una fuente de datos es creada, es automáticamente mantenida cada 5 minutos.

Interfaz

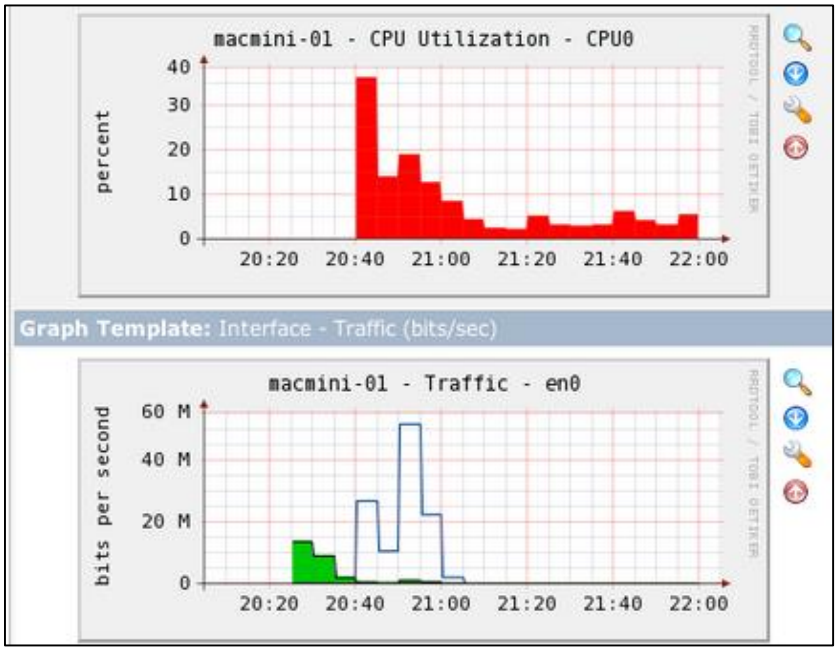


Figura 6: Ejemplo de interfaz CACTI[1]

La interfaz de CACTI presenta graficas de sencilla estructura y fáciles de leer, en las cuales principalmente se presenta la magnitud de cualquier variable a medir y se grafica contra el tiempo transcurrido entre mediciones, esta estructura sencilla es la que facilita en gran manera que el sistema sea capaz de ejecutarse en Hardware de bajas prestaciones de una manera eficiente.

[1] Fuente de imagen: Pagina web oficial de CACTI

1.4.2 PANDORA FMS

Pandora FMS es un software de monitorización orientado a todo tipo de entornos.

FMS son acrónimos de "Sistema de Monitorización Flexible" (en inglés). Su propósito es ser capaz de monitorizar tanto herramientas y sistemas de última generación -complejas-, como elementos anticuados, de difícil acceso y poca compatibilidad, en la misma plataforma.

Pandora FMS es un software de código abierto que sirve para monitorizar y medir todo tipo de elementos. Monitoriza sistemas, aplicaciones o dispositivos. Permite saber el estado de cada elemento de un sistema a lo largo del tiempo. 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.

Pandora FMS puede detectar si una interfaz de red se ha caído, un ataque de "defacement" en una web, una pérdida de memoria en algún servidor de aplicaciones, o el movimiento de un valor del NASDAQ. Pandora FMS puede enviar SMS si un sistema falla o cuando las acciones de Google bajan de 500 dólares.

Pandora FMS puede recoger información de cualquier sistema operativo, con agentes, específicos para cada plataforma, que recolectan datos y los envían al servidor. Hay agentes específicos para GNU/Linux, AIX, Solaris, HP-UX, BSD/IPSO y Windows 2000, XP, 7, 2003 y 2008.

Pandora FMS también puede monitorizar cualquier tipo de servicio TCP/IP, sin necesidad de instalar agentes, y monitorizar sistemas de red como balanceadores de carga, routers, switches, sistemas operativos, aplicaciones o impresoras si se necesita hacerlo de forma remota. Pandora FMS también soporta WMI para comunicarse directamente con sistemas windows de forma remota y SNMP para recolectar datos o recibir traps.

Algunos ejemplos de recursos comunes que pueden ser monitorizados con Pandora FMS son, la carga del procesador, el uso de disco y memoria, procesos que están

corriendo en el sistema, eventos determinados en un log, factores ambientales como la temperatura, la luz o la humedad, valores de aplicaciones como determinados textos en una página web, y en general cualquier cosa que se pueda recolectar de forma automatizada.

Pandora FMS está publicado bajo licencia GPL2 GNU General Public License, Pandora FMS es Open Source aunque dispone de una versión específica para empresas, bajo el modelo conocido como "openCore".

Interfaz

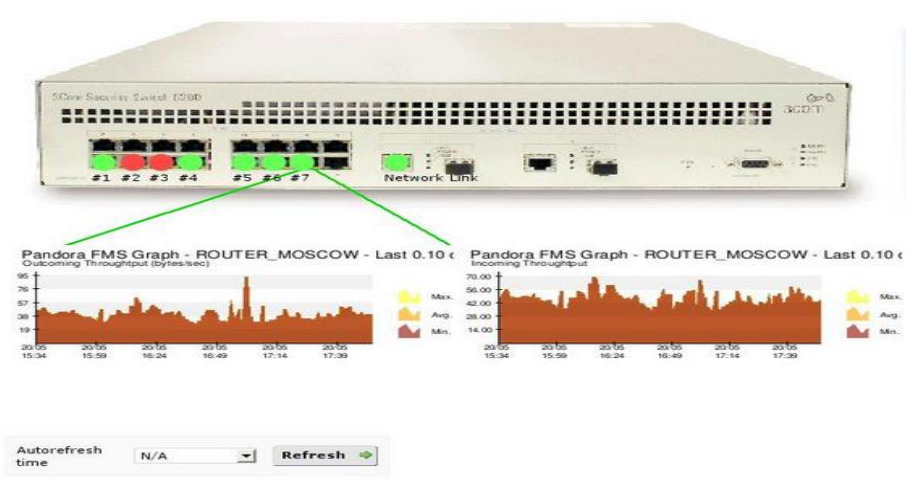


Figura 7: Ejemplo de Interfaz Pandora FMS[2]

La interfaz de Pandora nos permite de maneta significativa mejorar la calidad de visualización en tiempo real de los dispositivos brindando la capacidad de presentar una imagen específica relacionada con el dispositivo a monitorear, como el ejemplo de la figura 7, la cual nos presenta de manera grafica un router y el estado de las diferentes interfaces.

[2] Fuente de imagen: https://tuxedlinux.wordpress.com/2007/07/26/monitorizacion-de-servidores-con-pandora-fms/

1.4.3 NAGIOS XI

Nagios es un sistema de monitorización de redes ampliamente utilizado, de código abierto, que monitorea los equipos (hardware) y servicios (software) que se especifiquen, alertando cuando el comportamiento de los mismos no sea el deseado. Entre sus características principales figuran la monitorización de servicios de red (SMTP, POP3, HTTP, SNMP, etc.), la monitorización de los recursos de sistemas hardware (carga del procesador, uso de los discos, memoria, estado de los puertos, etc.), independencia de sistemas operativos, posibilidad de monitorización remota mediante túneles SSL cifrados o SSH, y la posibilidad de programar plugins específicos para nuevos sistemas.

Se trata de un software que proporciona una gran versatilidad para consultar prácticamente cualquier parámetro de interés de un sistema, y genera alertas, que pueden ser recibidas por los responsables correspondientes mediante (entre otros medios) correo electrónico y mensajes SMS, cuando estos parámetros exceden de los márgenes definidos por el administrador de red.

Nagios fue originalmente diseñado para ser ejecutado en GNU/Linux, pero también se ejecuta bien en variantes de Unix.

Interfaz

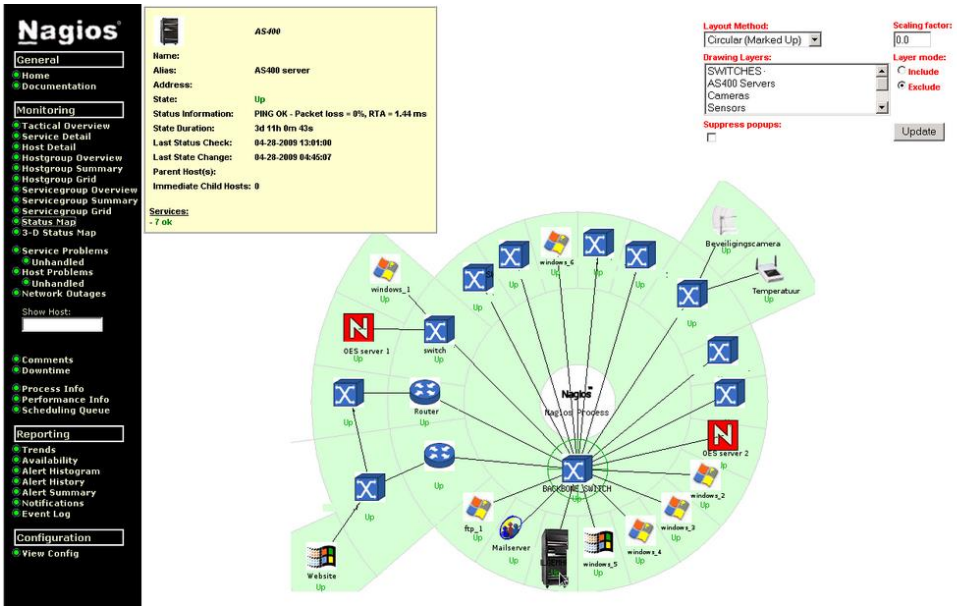


Figura 8: Ejemplo de interfaz Nagios XI[3]

[3] Fuente de Imagen: https://www.nagios.org/

1.4.4 PRTG

PRTG Network Monitor es la solución de monitorización de la compañía de monitorización de redes Paessler con una completa serie de características de monitorización, con una interfaz intuitiva y fácil de usar y tecnología de última generación, adecuado para redes de cualquier tamaño.

Figura 9: Ejemplo interfaz PRTG[4]

PRTG es una herramienta más vistosa y llamativa que permite la posibilidad de incluir una amplia gama de indicadores y estilos de gráficos para la presentación de las mediciones obtenidas, es una herramienta que resulta ser muy funcional sobre todo en la integración de dispositivos como equipos móviles o dispositivos de uso cotidiano como celulares o dispositivos móviles.

PRTG presenta entre sus principales inconvenientes la poca escalabilidad que el sistema presenta a causa de la limitada licencia proporcionada y la poca o nula compatibilidad con software creado a partir de terceros.

[4] Fuente de imagen: https://www.es.paessler.com/prtg

1.5 EQUIPOS OBJETIVOS

Las redes de telecomunicaciones modernas se basan en la arquitectura celular, dividiéndose la geografía en células o celdas, que quedan cubiertas a nivel radioeléctrico por estaciones base o BTS.

La principales funciones destacadas de las BTS son ofrecen un canal de broadcast que los terminales de abonado utilizan para medir el grado de cobertura disponible y tratar de cambiar a otra BTS si es preciso (handover); Ofrecen canales de tráfico para el establecimiento de llamadas telefónicas desde/hacia los terminales de abonado; Disponen de conexiones alámbricas o inalámbricas hacia las centrales telefónicas o BSC, desde donde se pueden encaminar las llamadas hacia otras zonas de la red.

Tomando como base estas estaciones, se consideran los equipos que comúnmente se pueden encontrar en ellas como objeto de estudio del presente trabajo.

1.5.1 ROUTER

Un router es un dispositivo que proporciona conectividad a nivel de red o nivel tres en el modelo OSI. 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 dispositivos IP que se pueden comunicar sin la intervención de un router (mediante conexiones de red), y que por tanto tienen prefijos de red distintos.

El funcionamiento básico de un router, consiste en enviar los paquetes de red por el camino o ruta más adecuada en cada momento. Para ello almacena los paquetes recibidos y procesa la información de origen y destino que poseen. Con arreglo a esta información reenvía los paquetes a otro router o bien al dispositivo final, en una actividad que se denomina 'enrutamiento'. Cada router se encarga de decidir el siguiente salto en función de su tabla de reenvío o tabla de enrutamiento, la cual se genera mediante protocolos que deciden cuál es el camino más adecuado o corto, como protocolos basado en el algoritmo de Dijkstra.

Por ser los elementos que forman la capa de red, tienen que encargarse de cumplir las dos tareas principales asignadas a la misma:

- Reenvío de paquetes: cuando un paquete llega al enlace de entrada de un router, éste tiene que pasar el paquete al enlace de salida apropiado. Una característica importante de los routers es que no difunden tráfico difusivo.

- Encaminamiento de paquetes: mediante el uso de algoritmos de enrutamiento tiene que ser capaz de determinar la ruta que deben seguir los paquetes a medida que fluyen de un emisor a un receptor.

Por tanto, debemos distinguir entre reenvío y enrutamiento. Reenvío consiste en tomar un paquete en la entrada y enviarlo por la salida que indica la tabla, mientras que por enrutamiento se entiende el proceso de hacer esa tabla.

En un router se pueden identificar cuatro componentes:

Puertos de entrada: realiza las funciones de la capa física consistentes en la terminación de un enlace físico de entrada a un router; realiza las funciones de la capa de enlace de datos necesarias para interoperar con las funciones de la capa de enlace de datos en el lado remoto del enlace de entrada; realiza también una función de búsqueda y reenvío de modo que un paquete reenviado dentro del entramado de conmutación del router emerge en el puerto de salida apropiado.

Entramado de conmutación: conecta los puertos de entrada del enrutador a sus puertos de salida.

Puertos de salida: almacena los paquetes que le han sido reenviados a través del entramado de conmutación y los transmite al enlace de salida. Realiza entonces la función inversa de la capa física y de la capa de enlace que el puerto de entrada.

Procesador de enrutamiento: ejecuta los protocolos de encaminamiento, mantiene la información de encaminamiento y las tablas de reenvío y realiza funciones de gestión de red dentro del enrutador.

Las interfaces comunes que se pueden encontrar en un Router son, puertos de comunicación serial, puertos Ethernet o FastEthernet, además de slots para tarjetas de comunicaciones como E1, T1.

Figura 10: Router CISCO[5]

[5] Fuente de imagen: http://www.cisco.com/

1.5.2 SWITCH

Es el dispositivo digital lógico de interconexión de equipos que opera en la capa de enlace de datos del modelo OSI. Su función es interconectar dos o más segmentos de red, de manera similar a los puentes de red, pasando datos de un segmento a otro de acuerdo con la dirección MAC de destino de las tramas en la red y eliminando la conexión una vez finalizada esta.

Los switch se utilizan cuando se desea conectar múltiples tramos de una red, fusionándolos en una sola red. Al igual que los puentes, dado que funcionan como un filtro en la red y solo retransmiten la información hacia los tramos en los que hay el destinatario de la trama de red, mejoran el rendimiento y la seguridad de las redes de área local (LAN).

Figura 11: Router Switch[6]

1.5.3 RADIOS MICRO ONDAS

Un radioenlace terrestre o microondas terrestre provee conectividad entre dos sitios (estaciones terrenas) en línea de mira (Line-of-Sight, LOS) usando equipo de radio con frecuencias de portadora por encima de 1 GHz. La forma de onda emitida puede ser analógica (convencionalmente en frecuencia modulada) o digital.

[6] Fuente de imagen: http://www.cisco.com/

Las microondas son ondas electromagnéticas cuyas frecuencias se encuentran dentro del espectro de las super altas frecuencias, SHF.

También se suele ofrecer por los instaladores de WiMAX para ofrecer servicio desde los lugares donde hay cobertura a aquellos cercanos en los que no la hay.

Las microondas se usan principalmente en sistemas de transmisión, sistemas inalámbricos, y también en transmisión de TV por cable, radares, navegación, posicionamiento, entre otros.

Los enlaces de microondas permiten llevar señales de un punto a otro. En un determinado punto, la señal se amplifica y se repite para dar mayor alcance y potencia a la señal.

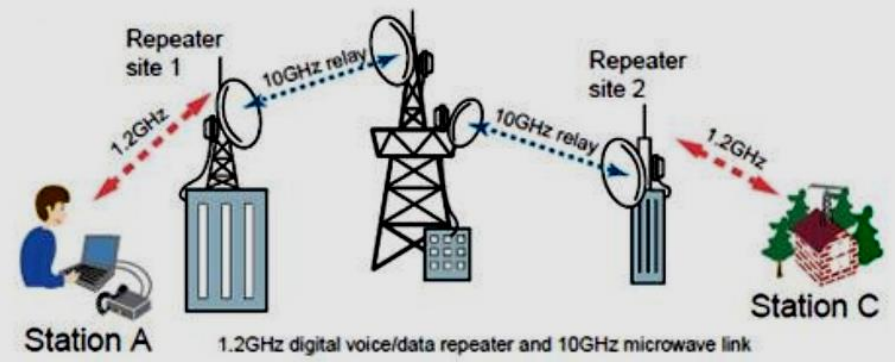


Figura 12: Ejemplo de funcionamiento de enlace microondas

Una estación Micro ondas se divide en dos partes:

ODU: Outdoor Unit – El Outdoor unit se compone por el receptor y transmisor.

IDU: Indoor Unit – El indoor unit es el equipo que se encarga de hacer la modulación, tiene partes de amplificación y enlaces en diferentes tipos de interfaz

1.5.4 INTERFACES DE ADAPTACIÓN

Se conoce como interface de adaptación a aquellas interfaces creadas con el fin de proporcionar capacidades de red a dispositivos que no cuentan con ella de manera nativa, con el fin de poder ser administrados o monitorizados.

Algunos de los dispositivos o componentes para los cuales es deseable que cuenten con la capacidad de medición de parámetros de operación y funcionamiento, pero sin embargo, por tratarse de equipo no pensado para poder se monitoreados o equipo antiguo no cuenta con esta función, son los siguientes:

UPS: Sistema de alimentación ininterrumpida, en inglés uninterruptible power supply (UPS), es un dispositivo que gracias a sus baterías u otros elementos almacenadores de energía, puede proporcionar energía eléctrica por un tiempo limitado y durante un apagón eléctrico a todos los dispositivos que tenga conectados. Otras de las funciones que se pueden adicionar a estos equipos es la de mejorar la calidad de la energía eléctrica que llega a las cargas, filtrando subidas y bajadas de tensión y eliminando armónicos de la red en el caso de usar corriente alterna.

Los UPS proporcionan energía eléctrica a equipos llamados cargas críticas, como pueden ser aparatos médicos, industriales o informáticos y de telecomunicaciones que, como se ha mencionado anteriormente, requieren tener siempre alimentación y que ésta sea de calidad, debido a la necesidad de estar en todo momento operativos y sin fallos (picos o caídas de tensión).

Figura 13: Ejemplo de UPS utilizado en equipos de telecomunicaciones[7]

[7] Fuente de Imagen: http://www.directindustry.es/prod/salicru/product-12333-442281.html

Equipos modernos de protección como el mostrado en la figura 13, a menudo cuentan con la integración de funciones de monitoreo remoto o gestión, pero esto representa una inversión mayor comparada con equipos normales, además, es común que estas funciones no existieran en equipos anteriores al año 2012, por lo cual se puede apreciar que la mayoría de estos equipos instalados actualmente en las redes de telecomunicaciones comerciales, no poseen la característica de monitoreo y gestión, por lo que es fundamental la integración mediante una interfaz adaptativa.

Aires acondicionados: El acondicionamiento de aire consiste en regular las condiciones en cuanto a la temperatura (calefacción o refrigeración), humedad, limpieza (renovación, filtrado) y el movimiento del aire dentro de los locales.

Entre los sistemas de acondicionamiento se cuentan los autónomos y los centralizados. Los primeros producen el calor o el frío y tratan el aire (aunque a menudo no del todo). Los segundos tienen un/unos acondicionador/es que solamente tratan el aire y obtienen la energía térmica (calor o frío) de un sistema centralizado. En este último caso, la producción de calor suele confiarse a calderas que funcionan con combustibles. La de frío a máquinas frigoríficas, que funcionan por compresión o por absorción y llevan el frío producido mediante sistemas de refrigeración.

Figura 14: Equipo de aire acondicionado para equipos de telecomunicaciones

A menudo las estaciones remotas de telecomunicaciones cuentan con equipo sensible y es necesario garantizar la climatización para los valores de óptimos de temperatura de los equipos teniendo en cuenta el calentamiento que estos generan por su operación normal, por lo cual es importante el monitoreo del funcionamiento de dichos dispositivos.

Plantas eléctricas de emergencia: Las plantas eléctricas de emergencias son maquinas que hacen mover a un generador con una fuerza mecánica, estos motores trabajan con diesel, comúnmente las plantas eléctricas de emergencia son utilizadas en lugares en los que es muy importante la que el servicio de energía eléctrica sea continuo, como en nuestro caso de estudio, pues es necesario garantizar una disponibilidad y un nivel de servicio constante.

Figura 15: Planta Eléctrica de emergencia

Una de las utilidades más comunes es la de generar electricidad en aquellos lugares donde no hay suministro eléctrico, generalmente son zonas apartadas con pocas infraestructuras y muy poco habitadas. Otro caso sería en locales de pública concurrencia, hospitales, fábricas, etc., que a falta de energía eléctrica de red, necesiten de otra fuente de energía alterna para abastecerse.

Es importante conocer los parámetros de generación de estas plantas, los niveles de voltaje, la corriente entregada, el tiempo de operación, el nivel de combustible restante, entre otras, son variables que deben ser controladas y censadas.

Otros dispositivos de interés, de los cuales se esperaría contar con una interfaz que permita su gestión y monitoreo son los siguientes:

- Sensores de variables físicas
- Baterías
- Luces de emergencia

Para efectos de esta tesis se utilizaran dos dispositivos para la generación de interfaces de adaptación, siendo una la tarjeta Flex Scada Q3, que permite el monitoreo de líneas trifásicas y monofásicas, tanto en corriente como en voltaje,

temperatura y voltaje DC, además para el monitoreo de variantes como temperatura ambiente, movimiento, humo entre otros, se utilizara el modulo Ethernet de la plataforma Arduino, este tema se presentara en el siguiente capítulo, a continuación se presentan las características de ambos dispositivos:

1.6 TARJETA FLEX Q3SCADA

La tarjeta de monitoreo y gestión presenta gran versatilidad para la creación de interfaces adaptativas, pues cuenta con entradas de conversión análogas digitales que permiten la publicación en el protocolo SNMP estudiado en secciones anteriores, con los cual se pueden integrar los equipos considerados en la sección anterior.

1.6.1 CARACTERÍSTICAS

Alimentación:

Voltaje de entrada: 8-30V DC

Entradas analógicas:

Canales: 8
Resolución: 24 bits
Frecuencia de muestreo: 16k SPS

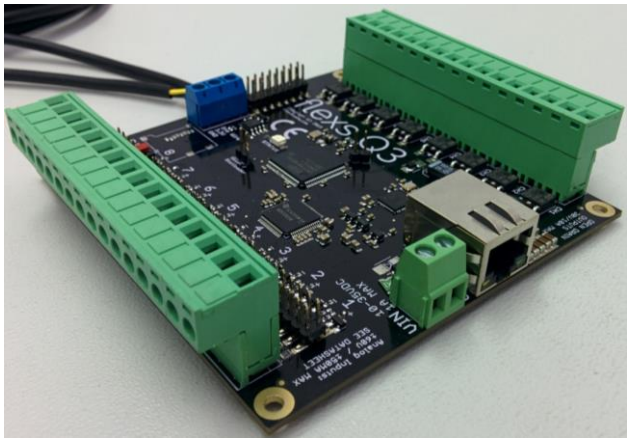


Figura 16: Tarjeta FlexQ3

1.6.2 APLICACIONES DE TARJETA FLEXQ3

Las características de la tarjeta que destacan para el uso práctico de la integración de interfaces adaptativas, es la flexibilidad de las ocho interfaces de entrada, que permiten diferentes funciones de medición para su posterior transmisión en el protocolo SNMP, dichas entradas se pueden ajustar para la medición de voltaje de corriente directa y alterna para diferentes rangos de tensiones, además la

adaptación de diferentes tipo de se sensores que permiten captar la mayoría de las variables necesarias para el monitoreo de estaciones remotas de telecomunicaciones.

Las salidas de estado sólido presentan una versatilidad considerable respecto a dispositivos actuadores como relés, para el control, en determinadas situaciones, su alta frecuencia de operación y la posibilidad de manejar grandes corrientes, son algunas de las características que destacan de esta tarjeta.

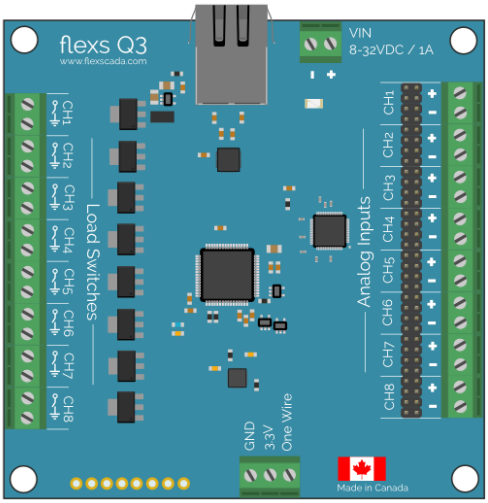


Figura 17: Esquema de entradas y salidas de tarjeta FlexQ3

Monitoreo de baterías o voltaje DC

La configuración de las entradas de la tarjeta, permiten realizar mediciones de corriente continua, para ser utilizada en diferentes aplicaciones, como la medición de la producción de energía en paneles fotovoltaicos o el almacenamiento de energía en bancos de baterías, mediante un voltaje de referencia se puede establecer un umbral que permita generar alarmas o active elementos actuadores dentro de la red.

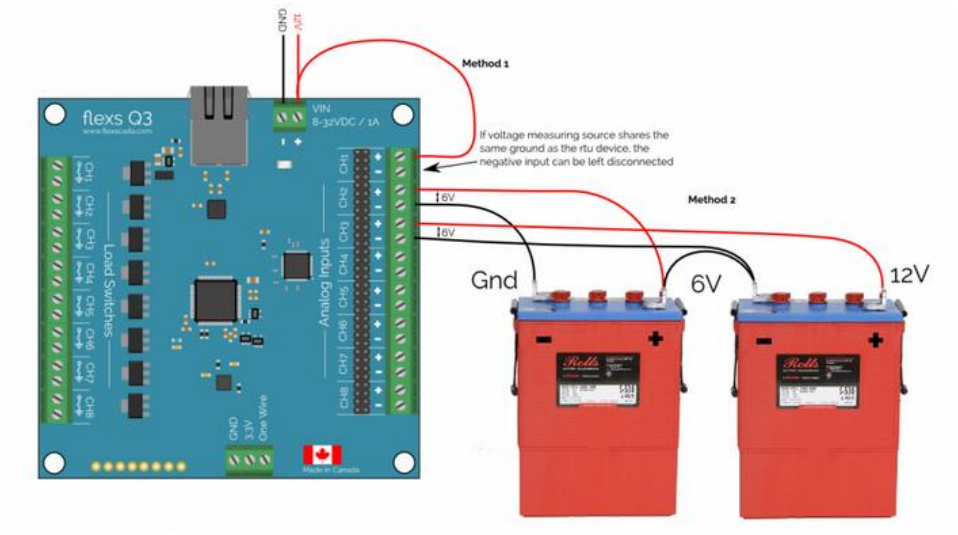


Figura 18: Medición de voltaje DC

Encendido y apagado de cargas remotamente

Utilizando las características de las salidas de estado sólido, es posible el encendido o apagado de motores con corrientes máximas de 10 Amperios, estas salidas pueden ser manipuladas mediante Traps del protocolo SNMP o como respuesta a una decisión tomada a partir de operadores lógicos y de comparación de las entradas.

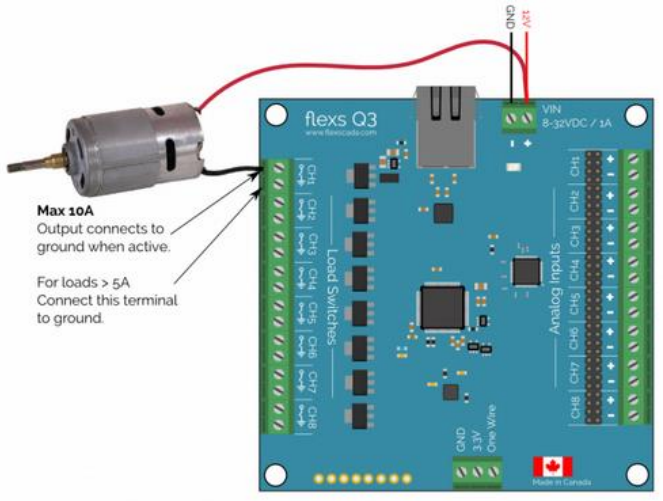


Figura 19: Encendido y apagado de motores de corriente continua

Sensores

Configurando las entradas de la tarjeta sea para la medición de pequeñas señales de corriente o voltaje, es posible integrar una amplia variedad de sensores, que

finalmente pueden ser calibrados mediante el software de administración asignándoles una ganancia o un Offset.

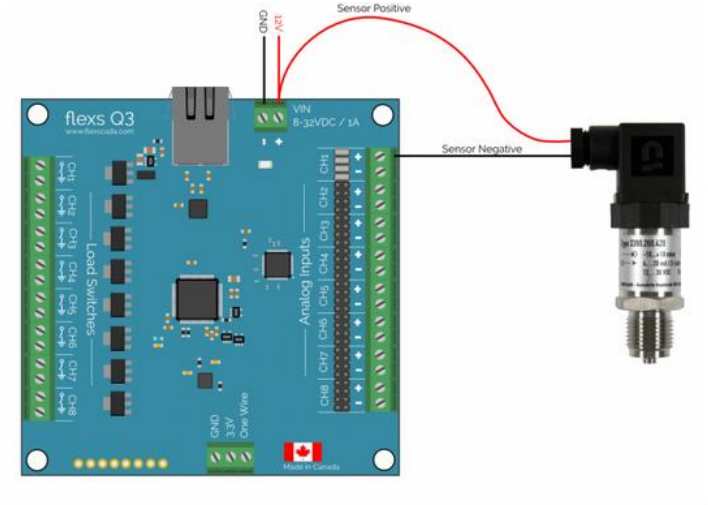


Figura 20: Uso de sensores para medición de variables ambientales y externad al sistema

Medición de Voltaje AC

La aplicación principal de la tarjeta de medición y monitoreo es la capacidad de monitoreo de voltajes trifásicos y monofásicos, con la adecuación de señal a partir de un divisor de tensión, la selección de la resistencia adecuada mediante el software de monitoreo permite que las mediciones a partir de generadores, tableros eléctricos y fuentes de alimentación, puedan ser publicadas mediante el protocolo SNMP.

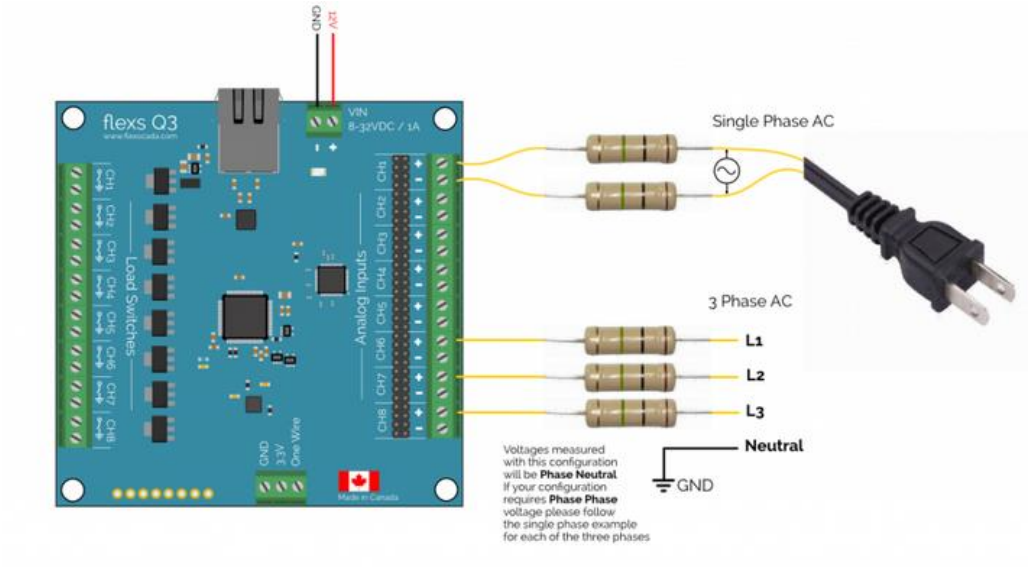


Figura 21: Esquema de medición monofásica y trifásica

CAPITULO II

2. ESTUDIO DE ALTERNATIVAS DE SOFTWARE-HARDWARE

En este capítulo se profundizara en el uso y características de los diferentes software y hardware de monitoreo específicamente en los seleccionados como candidatos para la implementación y pruebas en laboratorio, mencionados en el capítulo anterior, finalmente se seleccionara uno como propuesta de solución, además, se presentaran alternativas de hardware de monitoreo, que permitan a equipos que no poseen un protocolo de monitoreo en este caso SNMP, poder ser monitoreados, de una manera eficiente.

Los diferentes software de monitoreo presentados en este capítulo son muy variados en cuento a su desempeño y funciones, por lo cual se presentan las funciones destacadas de cada uno de ellos, desde su proceso de instalación hasta las acciones básicas para lograr monitorear diferentes tipos de equipos de telecomunicaciones, para finalmente realizar una comparación de los mismos y seleccionar una solución optima de monitoreo.

2.1 CARACTERÍSTICAS DE SOFTWARE DE MONITOREO

2.1.1 CACTI

Una de las principales ventajas de Cacti radica en la facilidad de utilizar un servidor ya empleado previamente u otros dispositivos de bajo costo, esto gracias a sus bajos requerimientos de sistema, la instalación consiste en la instalación de un servidor LAMP y los paquetes complementarios de CACTI y SPINE y RRDtool.

Proceso de instalación y configuración

Para las pruebas realizadas se instalara un servidor Ubuntu, sobre el cual correrá Cacti, seleccionando el Idioma y región correspondientes al servidor Ubuntu, el cual presenta los siguientes requisitos de sistema.

- Procesador x86 a 700 MHz.
- Memoria RAM de 512 Mb.
- Disco Duro de 5 GB (swap incluida).
- Tarjeta gráfica y monitor capaz de soportar una resolución de 1024x768.
- Lector de DVD o puerto USB.

La configuración de red del servidor debe ser estática, proporcionando una familia de IP y una máscara de sub red valida.

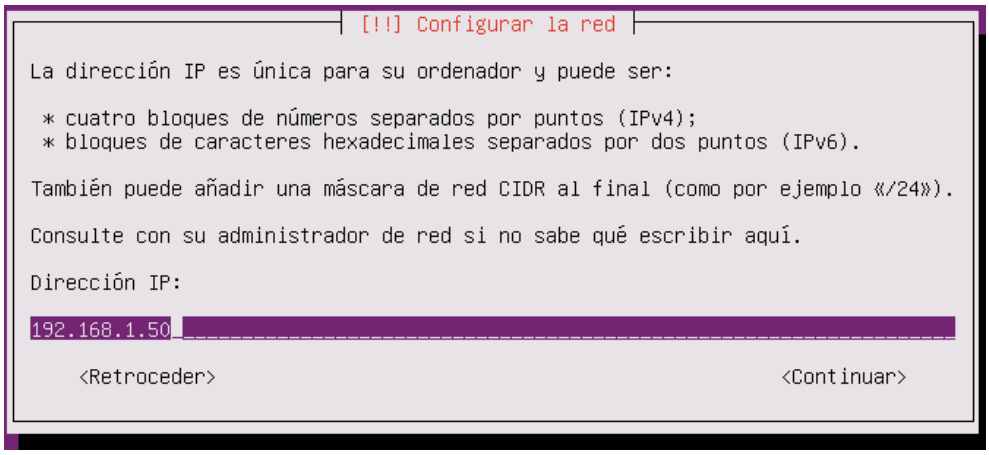


Figura 22: Configuración de IP estática en Ubuntu

Se debe configurar el Gateway del router de frontera que permita el enrutamiento fuera de la red local de comunicaciones, por definición esta IP corresponde a la primera IP valida dentro de la familia de red.

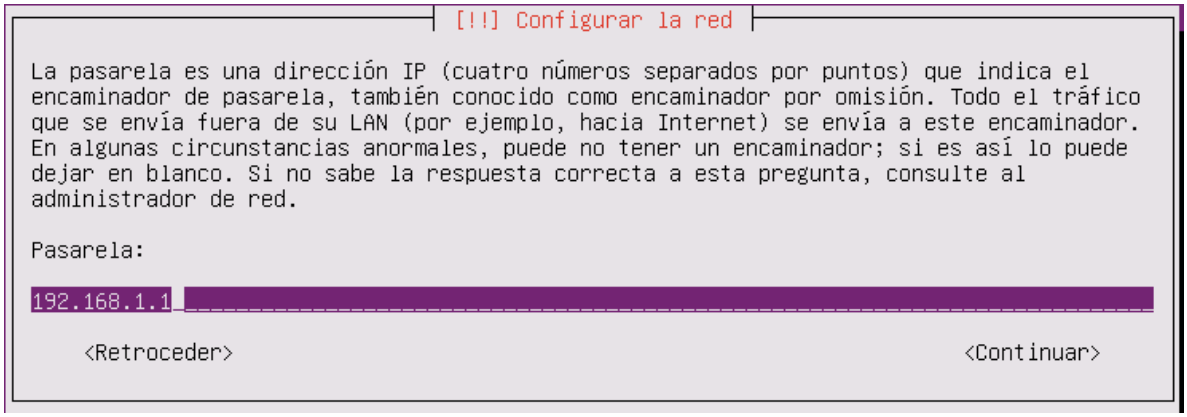


Figura 23: Selección de IP de Gateway

La selección de mascara de Subred depende de la red en la que se encuentre el servidor y las IP´s validas dentro de la red, para este caso se selecciona una máscara /24

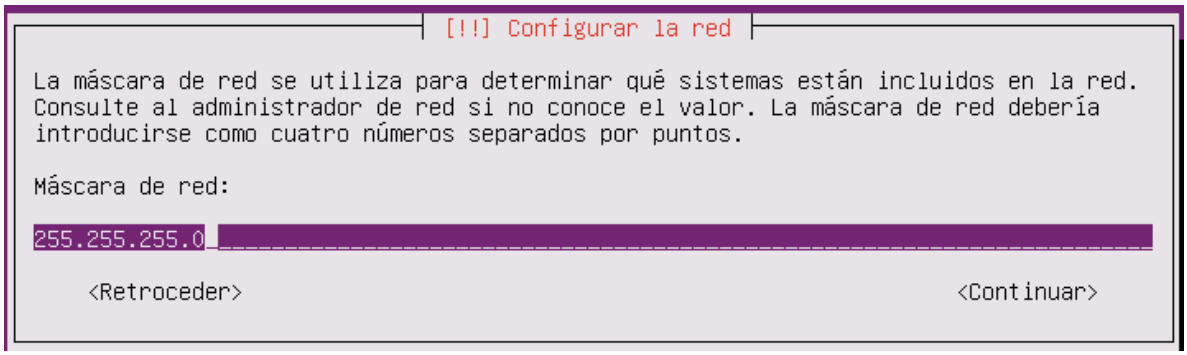


Figura 24: Configuraciones de mascara de Subred

Una vez correctamente instalado El servidor Ubuntu, la instalación de un servidor LAMP, es requerido, puesto que Cacti, la instalación de estos servicios se realiza mediante el gestor de paquetes por defecto Aptitude.

```
miguel@ubuntuCacti:~$
miguel@ubuntuCacti:~$
miguel@ubuntuCacti:~$ sudo aptitude install apache2 php5 mysql-server phpmyadmin
[sudo] password for miguel:
```

Figura 25: Instalación de servidor LAMP desde línea de comandos.

Posterior mente se instala la RRDtool que facilita la gestión y el bajo consumo de recursos para el uso de base de datos, como se explicó con anterioridad

```
miguel@ubuntuCacti:~$ sudo apt-get -y install rrdtool
```

Figura 26: Instalacion de RRDtool

Se requiere instalar los servicios de SNMP, y SNMPD en Linux para habilitar los servicios se transmisión de paquetes e información SNMP del servidor y el demonio SNMP para la monitorización requerida en tiempo real, respectivamente.

```
miguel@ubuntuCacti:~$ sudo apt-get -y install snmp snmpd_
```

Figura 27: Instalación de paquetes SNMP en Linux

Finalmente debemos instalar los paquetes requeridos de Cacti y spine, estos utilizaran el servidor apache y php para generar una interfaz web amigable con el usuario.

```
miguel@ubuntuCacti:~$ sudo apt-get -y install cacti cacti-spine
```

Figura 28: Instalación de paquetes de cacti y spine.

Una vez instalados los paquetes correspondientes, se debe reiniciar el servidor designado, con lo cual se encuentra completamente instalado el gestor de monitoreo, podemos tener acceso a su configuración mediante su interfaz de usuario web, ingresando en un navegador de internet la ip destinada al servidor de la siguiente manera:

Https://192.168.1.100/Cacti

De esta manera es posible configurar las diferentes características de Cacti y generar las diferentes graficas de las variables muestreadas.

A continuación se presenta una grafica generada para el uso de microprocesador de un router cisco:

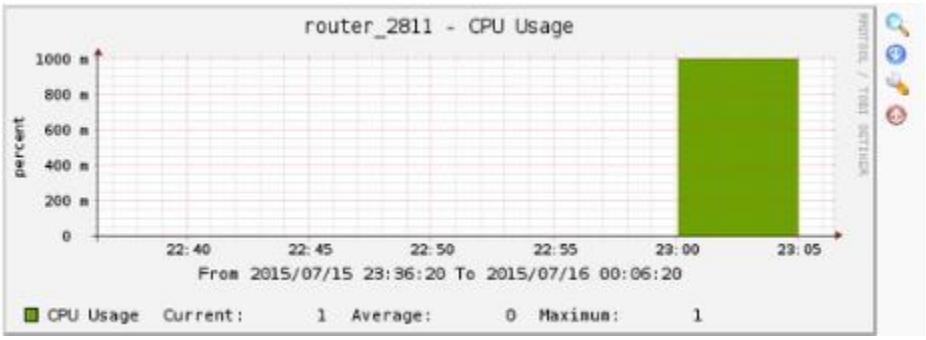


Figura 29: Monitoreo de CPU en servidor Ubuntu

Una opción practica de presentar la información de diferentes equipos conectados a la red, es la creación de arboles de gráficos en donde se agrupan de acuerdo a la información que presenta, o el tipo de grafico. Para generar este esquema a partir de la interfaz web, a continuación se presenta la configuración:

acceder a “Console > Management > Graph Trees”.

Figura 30: Ingresar a Creación de arboles de grafica

Seleccionamos “Default Tree”

Figura 31: Selección de Árbol

En la nueva ventana vemos el campo “Tree Items” que muestra todos los dispositivos y ramas del árbol de gráficas. Este árbol de gráficas se debe crear según interese.

En la pestaña “Tree Items” hacemos clic en “Add”.

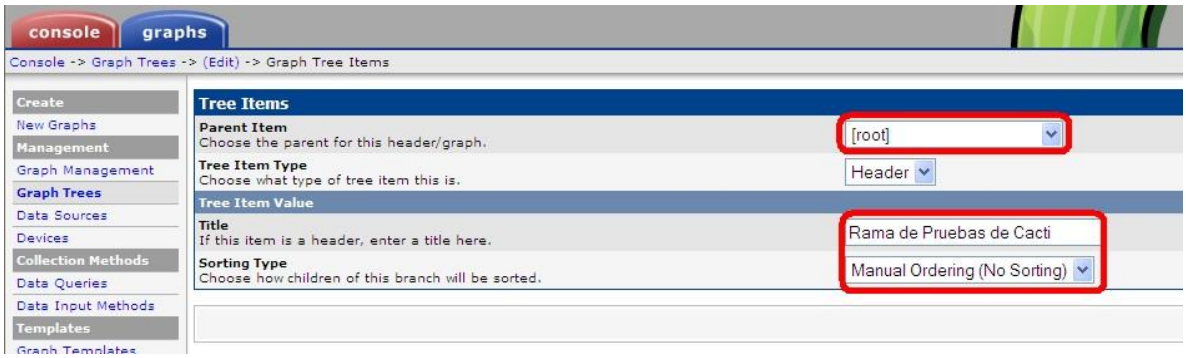


Figura 32: Creación de árbol

Como se mencionó anteriormente CACTI es capaz de correr en dispositivos de bajos recursos, tal es el caso de un computadora de bajas prestaciones como la Raspberry Pi modelo B, el cual es un ordenador de placa reducida o (placa única) (SBC) de bajo coste desarrollado en Reino Unido por la Fundación Raspberry Pi, con el objetivo de estimular la enseñanza de ciencias de la computación en las escuelas.

El diseño incluye un System-on-a-chip Broadcom BCM2835, que contiene un procesador central (CPU) ARM1176JZF-S a 700 MHz, un procesador gráfico (GPU) VideoCore IV, y 512 MB de memoria RAM.

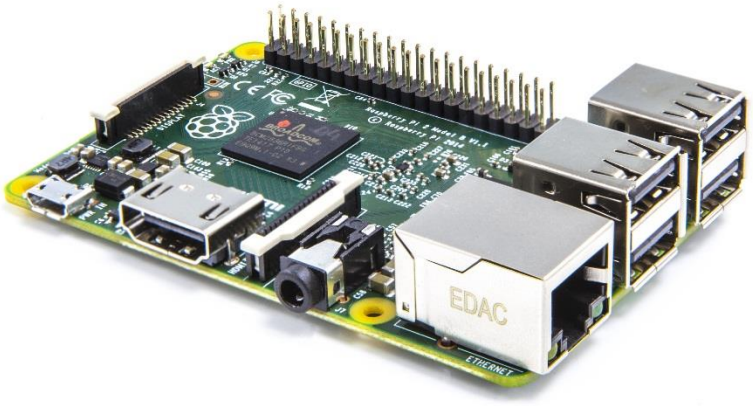


Figura 33: Raspberry Pi modelo B

Instalación en Raspberry pi

Para este caso cacti correrá sobre Raspbian Wheezy, un sistema operativo diseñado para este dispositivo basado en debían, como se mencionó anteriormente, se requiere un servidor LAMP para el uso de la interfaz web de Cacti, por lo que se utiliza Aptitude para su instalación, como se presenta a continuación:

```
# pt-get update
# apt-get install apache2
```

```
# apt-get install php5
# apt-get install mysql-client mysql-server
```

Se requiere además los complementos de PHP para my sql y SNMP, además de la Rrdtool, para el uso eficiente de las bases de datos.

```
apt-get install php5-mysql php5-snmp rrdtool snmp snmpd
```

Posterior a ello y a la configuración del servidor LAMP, se debe descargar y desempaquetar CACTI para lo cual utilizamos las siguientes instrucciones:

```
cd /var/www/
wget http://www.cacti.net/downloads/cacti-0.8.8a.tar.gz
tar xzvf cacti-0.8.8a.tar.gz
```

Configuración de bases de datos:

```
shell> mysql --user=root -p mysql
mysql> GRANT ALL ON cacti.* TO cactiuser@localhost IDENTIFIED BY 'cacti';
mysql> flush privileges;
```

Configuración de PHP:

```
$database_type = "mysql";
$database_default = "cacti";
$database_hostname = "localhost";
$database_username = "cactiuser";
$database_password = "cacti";
```

Finalmente Ingresar a la interfaz web, con lo cual podemos tener una versión funcional de CACTI, con una velocidad aceptable y la capacidad de medición de equipos de red, a bajo costo.

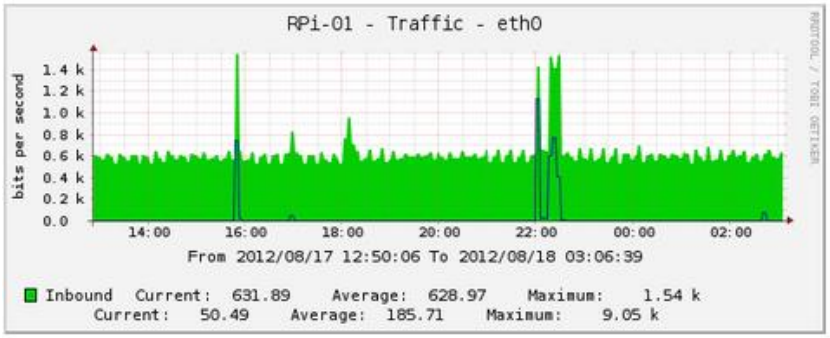


Figura 34: Monitoreo interfaz de red de Raspberry Pi.

2.1.2 NAGIOS

Nagios proporciona una licencia gratuita para pequeños entornos de monitoreo, con un máximo de 7 gestores, con la posibilidad de correr a partir de una máquina virtual, con lo cual se consiguen todas las características de su versión completa que puede ser adquirida a partir de los $3,500 con la posibilidad de monitoreo de 70 gestores.

Proceso de instalación y configuración

Se usara una distribución de CentOS 6.5 con una instalación mínima tal como nos aconsejan. La instalación de Nagios XI resulta ser sencilla. Solo se debe instalar desde un directorio en el que se tengan permiso de lectura y ejecución todos los usuarios, Para ello se usa como aconsejan en la documentación pertinente el directorio "/tmp."

Una vez establecido esto, debemos encontrar la imagen de dicho software y descomprimirla e intalarla con las siguientes instrucciones:

Para actualizar los repositorios de la distribución de Linux que se utilizara, se debe editar el archivo que contiene esta información:

```
# nano /etc/apt/sources.list
```

En caso de no contar con las fuentes siguientes, se deben agregar para su correcta actualización:

```
deb http://ftp.debian.org/debian/ squeeze-updates main contrib
deb-src http://ftp.debian.org/debian/ squeeze-updates main contrib
deb http://ftp.br.debian.org/debian/ squeeze main contrib non-free
deb http://ftp.debian.org/debian/ squeeze main contrib non-free
```

Posteriormente se debe descargar la imagen que contenga el sistema de monitoreo.

```
# wget http://sourceforge.net/projects/nagios/files/nagios-3.x/nagios-3.4.1/nagios-
3.4.1.tar.gz/download
```

Una vez descargado se procede a descomprimirlo e instalarlo

```
tar zxfv xi-2014r1.0RC3.tar.gz -C /tmp/
cd /tmp/nagiosxi
./fullinstall
```

Finalmente el servidor proveerá la siguiente información:

Nagios XI Installation Complete!
You can access the Nagios XI web interface by visiting:
http://192.168.1.49/nagiosxi/

Con lo cual podemos acceder a la interfaz web del sistema de monitoreo, mediante la IP estática que se le asigna al servidor Cen tOS, desde aquí continuaremos la configuración del software.

Es necesario proporcionar una URL y las credenciales de administración, como usuario y password, además de la zona horaria en la que se encuentre.

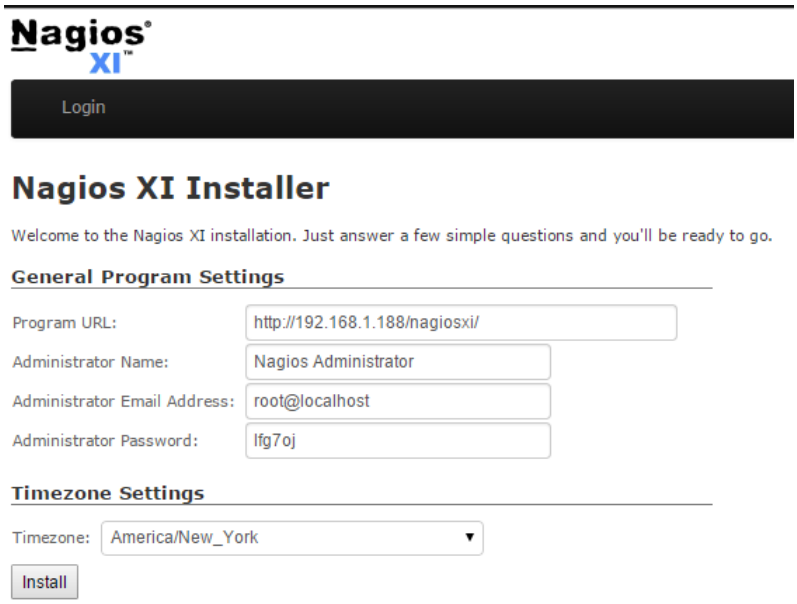


Figura 35: Configuración de interfaz usuario

Finalmente luego de completar el formulario los enviamos con el botón install lo cual termina la instalación del sistema completo de monitoreo y la creación de usuarios y passwords.

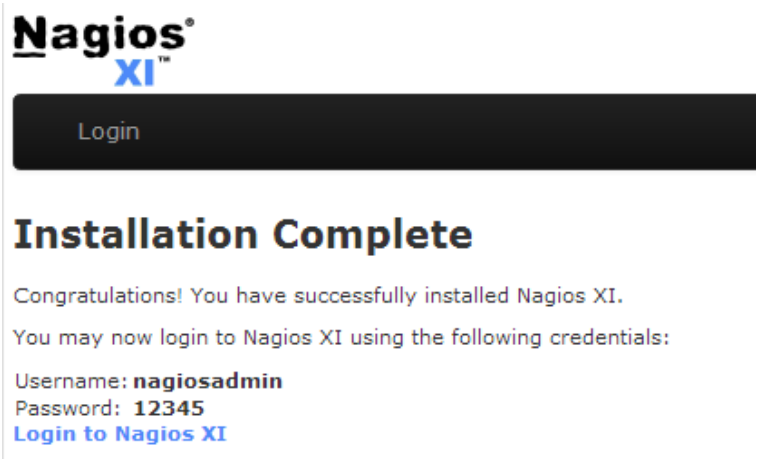


Figura 36: Confirmación de credenciales

Este software presenta la incorporación de un sistema de búsqueda automático de dispositivos, el cual facilita la detección y configuración de equipos a monitorear, también existe la posibilidad de configurar manualmente las interfaces a monitorear a partir de sus MIB's, la configuración de usuario es intuitiva y fácil de gestionar, a continuación se presenta la incorporación de graficas de uso y eficiencia del dispositivo de frontera de la red de telecomunicaciones y el servidor desde el cual se procesa la información.

Monitoreo de interface

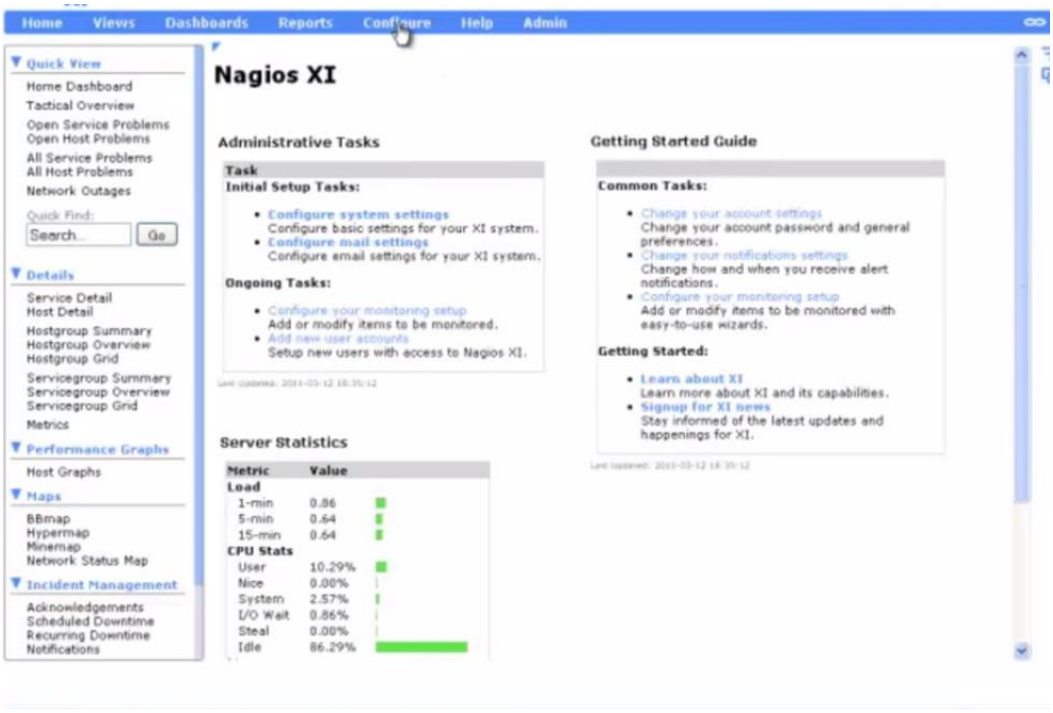


Figura 37: Panel principal NAGIOS

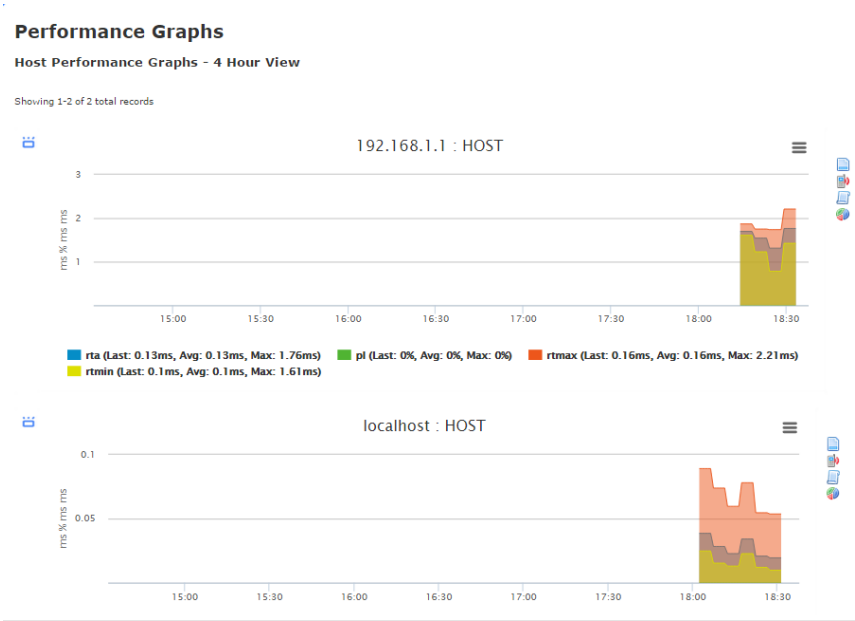
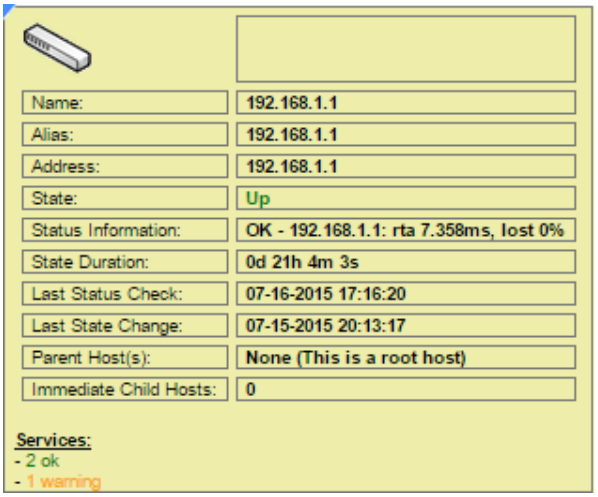


Figura 38: Monitoreo Interfaz Router con Nagios

Una de las principales funciones y más llamativas de este software es la capacidad de presentar mapas de redes generados automáticamente a partir de los sistemas que el Software este monitoreando, con esto de tiene una idea más específica de la red correspondiente, como ejemplo se presenta los dos casos expuestos a continuación, donde se denota la inteconexion existente entre el servidor de datos Nagios y un dispositivo de red existente en la topología mostrada.

Figura 39: Creación de mapas de red en Nagios

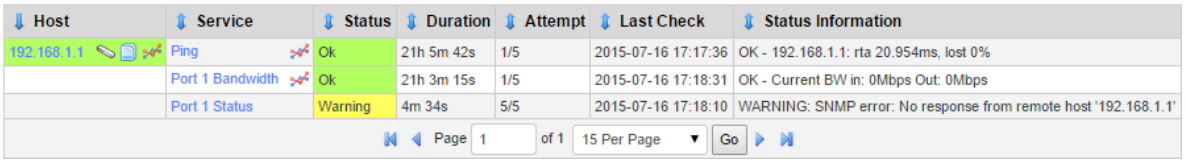
También es importante la capacidad de presentar información útil del estado de las interfaces y dispositivos a partir de los mapas de topología antes mencionados, en este caso, bastaría con seleccionar un dispositivo existente en la red de telecomunicaciones y automáticamente se genera dicha información útil para la planificación de la red y el checkeo de los dispositivos base.



Name:	192.168.1.1
Alias:	192.168.1.1
Address:	192.168.1.1
State:	Up
Status Information:	OK - 192.168.1.1: rta 7.358ms, lost 0%
State Duration:	0d 21h 4m 3s
Last Status Check:	07-16-2015 17:16:20
Last State Change:	07-15-2015 20:13:17
Parent Host(s):	None (This is a root host)
Immediate Child Hosts:	0

Services:
- 2 ok
- 1 warning

Figura 40: Información de dispositivos de Red



Host	Service	Status	Duration	Attempt	Last Check	Status Information
192.168.1.1	Ping	Ok	21h 5m 42s	1/5	2015-07-16 17:17:36	OK - 192.168.1.1: rta 20.954ms, lost 0%
	Port 1 Bandwidth	Ok	21h 3m 15s	1/5	2015-07-16 17:18:31	OK - Current BW in: 0Mbps Out: 0Mbps
	Port 1 Status	Warning	4m 34s	5/5	2015-07-16 17:18:10	WARNING: SNMP error: No response from remote host '192.168.1.1'

Page 1 of 1 15 Per Page Go

Figura 41: Información de estado dispositivo de red

Interfaz de usuario web de gran alcance: Una interfaz gráfica de gran alcance le permite personalizar el diseño, la disposición y las preferencias de cada usuario, proporcionando al personal de TI la flexibilidad que ellos buscan. El uso de la tecnología AJAX permite actualización en tiempo real del estado de los elementos de infraestructura monitoreados.

Gráficos de análisis de rendimiento y planificación: La integración de gráficos automáticos para el análisis de tendencias y la planificación permiten planear la actualización de los sistemas antes de que se conviertan en obsoletos. Los gráficos

se crean automáticamente al configurar Nagios XI para monitorear un nuevo elemento. El uso de la tecnología RRD para la realización de los gráficos permite una utilización óptima de almacenamiento donde se guardan los datos.

Dashboards: Un panel de control proporciona un vistazo a toda la informaciones proporcionadas por el sistema de monitoreo y las informaciones proporcionadas por terceros. El uso de dashlets permite a los usuarios de personalizar su propio panel de control con todos los datos que se consideren más útiles.

Vistas: Proporcionar a los usuarios un acceso rápido a las informaciones que les resulten más útil. La capacidad de configurar vistas de manera cíclicas garantizar al personal de TI para obtener acceso a la información crítica rápidamente.

Configuración gráfica vía Web: a través de la interfaz de configuración basada en web, los administradores pueden conceder fácilmente a otros miembros del personal un control completo del sistema de monitoreo, o sólo las configuraciones o ajustes del sistema y mucho más.

Wizards de Configuración: A través de sencillos wizards cada usuario puede añadir nuevos dispositivos, servicios y aplicaciones, todo sin tener la necesidad de conocer complicadas nociones de monitoreo o modificar algun fichero de configuración. La capacidad de crear guía de procedimiento permite a las organizaciones de adaptar Nagios XI a todas las necesidades.

Gestión Avanzada de la configuración: Con la adopción de una interfaz web avanzada, Nagios Core permite a los administradores un control completo sobre toda la configuración del sistema de monitoreo. Gracias a las funciones de importación, puede migrar los archivos de configuración de Nagios Core a Nagios XI.

Administración avanzada de usuarios: Facilita la gestión del sistema de monitoreo Nagios XI, simplemente mediante la configuración de los usuarios. Añadir un nuevo usuario es un proceso muy simple, sólo unos pocos clics y nuevos usuarios recibirán un correo electrónico con sus datos de acceso.

Notificaciones personalizadas: los usuarios pueden gestionar el tipo de notificaciones que quieren recibir, cómo recibir las notificaciones y el tipo de información contenida en las mismas notificaciones.

Operaciones sin parar: Protección contra el fallo del sistema de monitoreo causada por errores de configuración o errores de usuario. La introducción de nueva funcionalidad Non -Stop Operations Manager asegura la continuidad del servicio

Arquitectura modular: Nagios XI se construye utilizando componentes de código abierto probadas, para adaptarse a cualquier necesidad de las empresas. Cientos de add-ons y scripts desarrollados por la Comunidad de desarrolladores, permiten extender la funcionalidad básica de Nagios XI. El diseño modular permite una fácil personalización de la solución..

Base de datos: Mediante el uso de una base de datos integrada y la introducción de APIs, se puede acceder a los datos de monitoreo. La implementación de estrategias para la autenticación integrada para proteger los datos sensibles, manteniéndolos a salvo de accesos no autorizados y permitir a las organizaciones a desarrollar su propio front-end para mostrar los datos de vigilancia

2.1.3 Pandora FMS

La plataforma oficial de Pandora FMS es Linux. Desde la version 5.1 también se soporta Windows Server. Oficialmente se soportan, para el servidor y la consola, las siguientes versiones: Windows Server (2003 o superior), RedHat Enterprise (RHEL) 6.x, CentOS 6.x, SLES 11 SP1 o superior, OpenSUSE 11.x o superior, Debian 5.x o superior, Ubuntu 11 o superior.

Requisitos de Base de datos

Antes de comenzar a instalar Pandora FMS, se necesita tener un servidor de MySQL funcionando (Oracle y PostgreSQL se soportan, pero aun de forma experimental). Esto significa que antes de instalar Pandora, se necesita tener corriendo, bien configurado y operativo, el software de base de datos MySQL, puede estar en el mismo servidor físico donde se quiere ejecutar Pandora FMS, o en un servidor independiente, de forma que la consola y el servidor, accedan a él a través de la red, via TCP/IP. En resumen, se necesitará:

- Dirección IP del MySQL Server, o 'localhost' si se instala en el mismo servidor de Pandora.
- Usuario con privilegios para crear bases de datos y usuarios (generalmente root). Este usuario debera poderse conectar desde la IP del servidor donde instalemos pandora fms.
- Password del usuario con privilegios

Requisitos para el agente

El agente puede ejecutarse en cualquier hardware que pueda ejecutar el sistema operativo mínimo requerido, que es:

Componente	Sistema Operativo
Pandora Agent 4.0 o superior	RedHat Enterprise (RHEL) 6.x CentOS 6.x SLES 11 SP1 o superior OpenSUSE 11.x o superior Debian 5.x o superior. Ubuntu 11 o superior. HPUX B.11.11 o superior, con Perl 5.8. AIX 4.3.3 o superior, con Perl 5.8. Sistemas BSD (NetBSD, OpenBSD, FreeBSD), con Perl 5.8. MacOSX 10.6 or higher. Solaris 8 o superior, con Perl 5.8. Windows NT4 (ver notas especiales de esta version). Windows XP Windows 2000 Windows 2003 Windows 2008 Windows 7 Windows 8

Figura 42: Informacion de consulta por hardware y software usado

Aunque puede trabajar sobre cualquier sistema operativo con Perl 5.8 instalado y con iThreads habilitados, se recomienda y está soportado únicamente sobre Linux y FreeBSD. También funciona sobre sistemas Solaris.

Hay que destacar que Pandora FMS necesita un servidor MySQL para almacenar toda la información. Este servidor puede instalarse en cualquier plataforma soportada por MySQL (Windows, Linux, Solaris, etc).

Se deberá instalar Perl 5.8, al menos, para que el servidor funcione correctamente. Además de los paquetes de SNMP del sistema operativo (net-snmp) para usar el servicio SNMP de Pandora FMS. También se requiere una base de datos (MySQL). También se requieren los paquetes nmap y opcionalmente el paquete xprobe2 para utilizar las características avanzadas de reconserver, así como las bibliotecas traceroute de Perl para poder hacer autodescubrimientos de red. Por último, también es necesario, el cliente binario de WMI para hacer consultas WMI contra sistemas Windows. Dicho cliente binario es parte del proyecto SAMBA (v4) y puede ser compilado -no sin cierta dificultad- en cualquier entorno Unix. La version

enterprise requiere un binario que hay que compilar, y para el cual existen versiones de todas las plataformas oficialmente soportadas.

Requisitos para la consola: De igual manera que el servidor, se recomienda su operación sobre sistemas Linux, pero dado que la interfaz web es una aplicación AMP pura (Apache, MySQL y PHP), podría trabajar teóricamente sobre cualquier sistema que lo soporte: Windows, Unix, etc.

Requisitos para administrar la herramienta via WEB: Se deberá disponer de un navegador web para instalar y comprobar el funcionamiento de la consola. En principio no se requiere que el navegador tenga el complemento de FLASH instalado, aunque se recomienda para poder hacer uso de las gráficas interactivas en Flash.

Dependencias de paquetes

Pandora FMS depende en gran parte del sistema operativo Linux, pero además necesita paquetes adicionales que muchas veces no vienen instalados de forma predeterminada. En el proceso de instalación se detallan de forma específca esas dependencias para sistemas Debian/Ubuntu y OpenSUSE.

Cuestiones previas a la instalación

MySQL

Se necesitará un servidor MySQL operativo antes de instalar Pandora, ya que el siguiente paso tras instalar los paquetes de Pandora, es configurar el acceso a la BBDD de datos. Si está instalando Pandora FMS a la vez que el servidor MYSQL, recuerde que tiene que arrancar y configurar el acceso al usuario root de MySQL. Esto se hace mediante dos comandos:

1. Arrancar:

```
/etc/init.d/mysql start
```

2. Configurar el password de root

```
mysqladmin password <password>
```

Donde '<password>' es el password que establece para el usuario root. Este password se le pedirá en el proceso de instalación de Pandora FMS.

Oracle

Se necesita comprar el producto de base de datos de Oracle y tenerlo funcionando en tus sistemas antes de instalar Pandora.

Servidor y Consola

La maquina donde este Pandora Consola y la maquina donde este Pandora Servidor que usualmente suelen ser las mismas, tienen que tener acceso por red a la máquina de BD de Oracle por el puerto TCP 1521.

Para las maquinas con Pandora Servidor o Pandora Consola es necesario tener:

En maquinas de 64 bits y Suse/CentOS/RedHat/Mandriva instalar los siguientes paquetes

- oracle-instantclient12.1-basic-12.1.0.2.0-1.x86_64.rpm
- oracle-instantclient12.1-devel-12.1.0.2.0-1.x86_64.rpm
- oracle-instantclient12.1-sqlplus-12.1.0.1.0-1.x86_64.rpm

Exportar estas variables de entorno para instalar los siguientes elementos:

- export ORACLE_HOME="/usr/lib/oracle/12.1/client64"
- export LD_LIBRARY_PATH="/usr/lib/oracle/12.1/client64/lib"

Consola

Para Pandora Consola que funciona con PHP es necesario:

- Instalar la libreria OCI8 con el gestor de repositorios de módulos compilados de PHP PECL (normalmente se instala con PHP si no en tu sistema operativo lo podras instalar).

```
$ sudo pecl install oci8
```

Definir ocalización de las librerías:

```
instantclient,/usr/lib/oracle/12.1/client64/lib
```

Cuando pregunte por ORACLE_HOME se escribe all y luego escribe la dirección raíz:

instantclient,/usr/lib/oracle/12.1/client64/lib

Y en el fichero de configuración de PHP (normalmente php.ini en alguna parte del /etc de la maquina) para que use el módulo de Oracle para PHP, añadiendo la linea:

extension=oci8.so

Servidor

En el servidor al estar escrito en Perl es necesaria la librería para este lenguaje.

La puedes descargar de la web de CPAN en:

DBD::Oracle

Orden de instalación de Pandora FMS

Es recomendable seguir el siguiente orden al instalar Pandora FMS:

- Instalar la consola
- Instalar el servidor

La razón es que la base de datos MySQL que usa el servidor se crea en el proceso de configuración inicial de la consola, y por ello para asegurar el correcto funcionamiento del servidor es recomendable realizar primero el proceso de instalación completo de la consola.

Además no es necesario que la consola y el servidor de Pandora FMS se encuentren alojados en la misma máquina, ya que es posible indicarle al servidor dónde se encuentra la base de datos MySQL mediante el archivo de configuración del servidor.

La instalación del agente la podemos realizar sin ningún problema antes o después de instalar el servidor y la consola ya que es independiente de estos y puede estar instalado en cualquier máquina.

Instalación de la versión Enterprise de Pandora FMS

A partir de la versión 4.0.2 el numero de licencia de Pandora FMS habilita el uso de las características Enterprise. Esto significa que si tiene instalada una version enterprise 4.0.2 sin un número de licencia válido, no funcionará. Debe instalar primero la version OpenSource, meter el nº de licencia y luego instalar la version enterprise en ese orden. Como resumen:

- Instale la consola OpenSource.

- Acceda a la consola, vaya a la seccion de setup e introduzca su licencia ahí (ver imagen más abajo)
- Instale la consola de la versión Enterprise

No obstante, si lo hace en otro orden, en la propia pantalla que le notificará el error podrá introducir su licencia. La validación de la licencia solo se realiza en la consola. No en el servidor.

Introducir la licencia, en la opcion principal de configuration (setup):

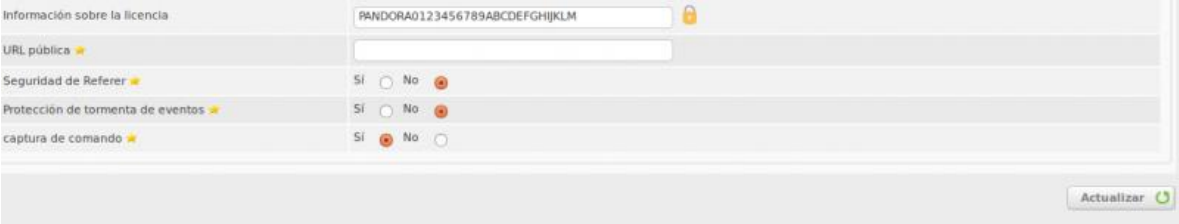


Figura 43: Introducir Licencia de Pandora

A partir de la versión 5.X, se ha mejorado la seguridad en la licencia y este no es el método correcto. Para instalar la licencia en la versión 5, hay que instalar primero la consola enterprise y acceder a la pantalla de login. Una vez hayamos accedido nos encontraremos con la siguiente pantalla: Activar licencia v5.X

Añadir agente para monitorizar switch Cisco en Pandora FMS

Accederemos a la consola de administración de Pandora FMS, en el menú lateral izquierdo accederemos a "Administración" - "Gestionar la monitorización" - "Gestionar agentes", pulsaremos en "Crear agente":

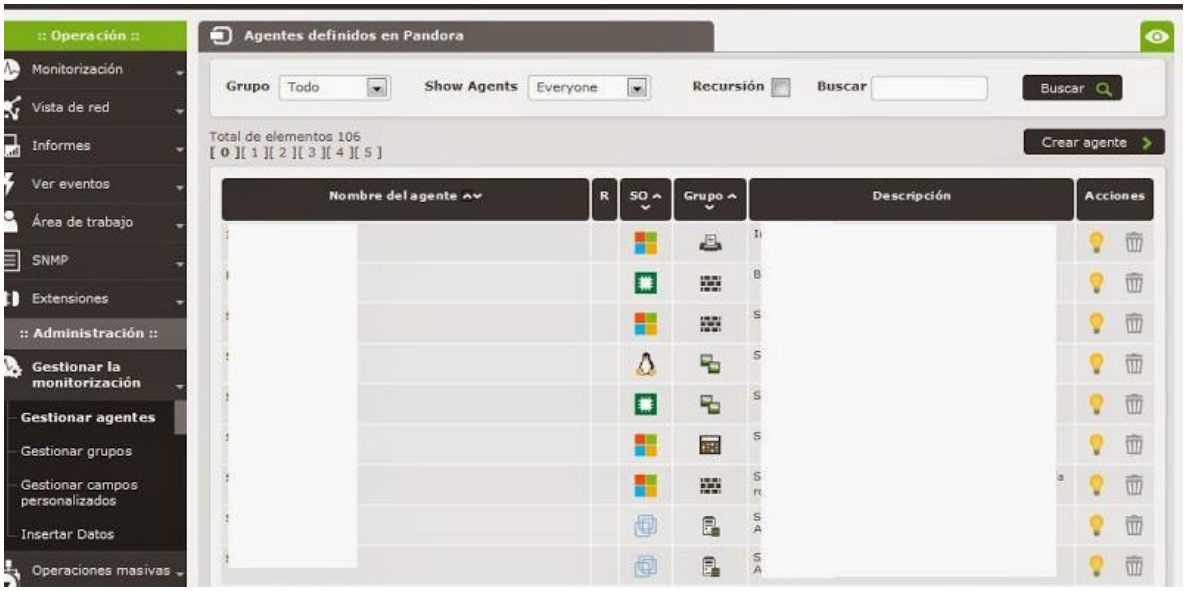


Figura 44: Creación de agentes en Pandora

Introduciremos los datos para el nuevo agente (servidor, equipo, impresora), al menos:

Nombre del agente: nombre identificativo del switch, en este caso, por ejemplo Switch_001.

Dirección IP: en agentes que son servidores y equipos, si en nombre se especifica el nombre DNS o hostname, no es necesario especificar la dirección IP, aunque sí es recomendable. Para el caso de agentes como un switch sí es muy recomendable especificar la IP.

Grupo: si hemos definido grupos, aquí podemos asignar el agente a un grupo, por ejemplo "Red".

Servidor: seleccionaremos el servidor de Pandora FMS.

SO: podemos seleccionar "Cisco" para que nos muestre el icono correspondiente en los agentes de switch.

Descripción: podemos especificar aquí una descripción para identificar el switch (lugar donde está instalado, nombre, uso, etc.).

Tras introducir los datos del nuevo agente pulsaremos en "Crear":

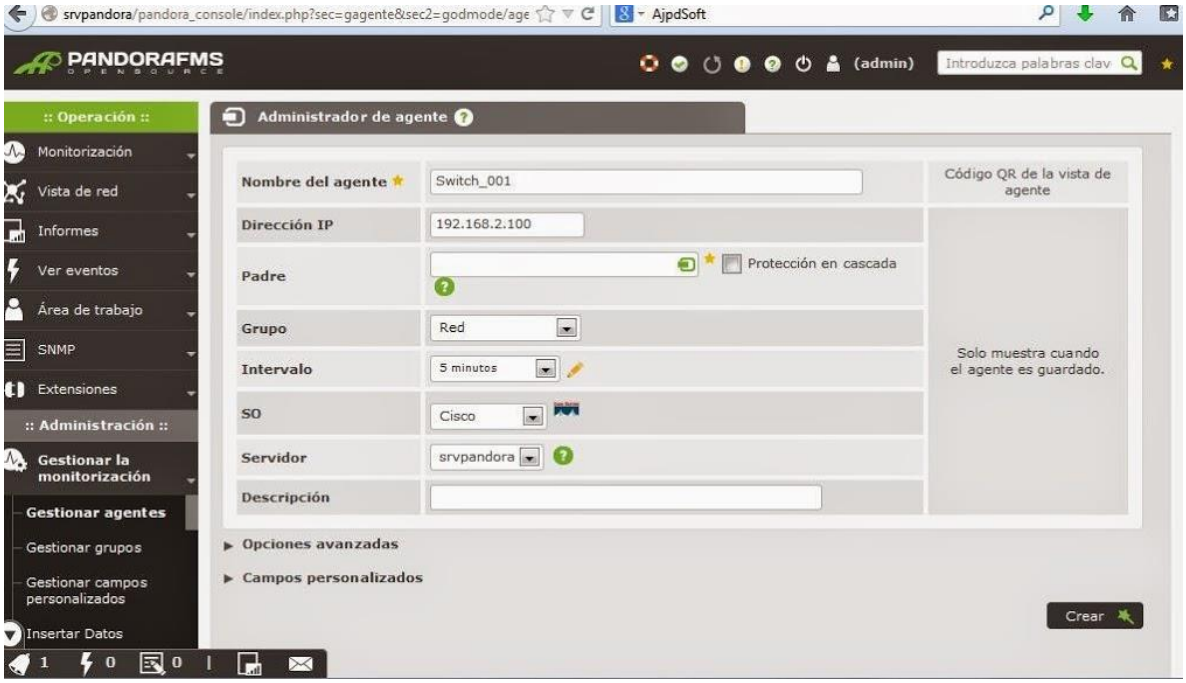


Figura 45: Cracion de Nuevo Agente en Pandora

2.2 COMPARACIÓN ENTRE SOFTWARE DE MONITOREO

Finalmente basándose en las pruebas de laboratorio se busca realizar la comparación del diferente software estudiado para proponer una solución a desarrollarse, en esta sección se profundizara en el detalle de las características, precios y funciones de cada uno de los sistemas de monitoreo propuestos.

2.2.1 PRECIO

Pandora FMS	NAGIOS	CACTI	PRTG
Versión OpenSource: gratuita hasta 100 nodos **Version Enterprice:** depende de los nodos utilizados	**Versión OpenSource:** gratuita hasta 70 gestores **Version Standard:** $1,995 hasta 100 gestores	**Versión OpenSource:** Gratuita	**Versión OpenSource:** gratuita hasta 100 nodos **Version Enterprice:** depende de los nodos utilizados

Tabla 1: Comparativa de precios de sistemas de monitoreo

Mejoras en la versión de pago

Pandora: Monitorización de servicios, Mejoras de optimización para grandes entornos, mayor almacenamiento en base de datos de registros históricos, gestión remota de agentes, personalización de informes.

Nagios: capacidad de implementación de reportes automáticos, mejora en los asistentes de configuración y detección de hardware, presentación de Dashboard mejorada, precio máximo de $4,995 para gestores ilimitados

PRTG: soporte técnico ilimitado, incremento de sensores soportados, capacidad de almacenamiento en la nube.

2.2.2 CARACTERÍSTICAS

Características	Pandora FMS	Nagios	Cacti	PRTG
Generación de graficas	Si	Si	Si	Si
Grupos lógicos de Equipos	Si	Si	Si	Si
Predicción de estadísticas	Si	Si	No	No
Compatibilidad con dispositivos móviles	Si	No	No	No

Características	Pandora FMS	Nagios	Cacti	PRTG
Escalabilidad	Requiere re evaluación y nueva implementación	Fácil y sostenible	Depende de recursos del host	Aumento de inversión considerable
Acceso a desarrollo de aplicaciones	Difícil	Fácil	Fácil	Difícil
Uso de recursos de servidor	Alto	Bajo	Bajo	Alto
Generación de reportes	Si	Si	No	Si
Requiere agentes	Con Agentes y Sin Agentes	Si	Si	Si

Tabla 2: Comparativa de sistemas de monitoreo de acuerdo a sus características

Basado en las características, las posibilidades de la versión gratuita y el precio de compra de licencias para los cuatro software en cuestión, en la implementación de una solución se presentara NAGIOS como principal candidato, pues brinda una interfaz más amigable al usuario con pre configuraciones para equipos de red más comunes, presenta un mayor grado de posibilidad de escalamiento y mayor robustez en su versión de Open Source, presenta la capacidad automática de generación y detección de software para creación de mapas de topología de red.

Cacti presenta una versatilidad destacable en la implementación de una estación de monitoreo con pocos recursos, es un software que puede correr en dispositivos como la Raspberry pi u otras computadoras de bajas prestaciones, siendo estable y confiable.

CAPITULO III

3. IMPLEMENTACIÓN DE RED DE MONITOREO BASADA EN NAGIOS E INTEGRACIÓN DE INTERFACES NO NATIVAS EN PROTOCOLO SNMP

Para efectos de las pruebas de laboratorio se utilizara un circuito representativo de un esquema de telecomunicaciones básico, el cual que contenga los elemento comunes encontrados en dichas redes, como Routers, Switchs, Teléfonos IP, Computadoras, Servidores, los cuales por si mismos son nativos del protocolo de monitoreo SNMP, pero adicionalmente se integrara la medición de variables que no son nativas de dicho protocolo como el voltaje de un tablero eléctrico, la humedad relativa y temperatura ambiente del sitio donde se encuentra ubicado el equipo de telecomunicaciones, de igual manera la carga de sistemas de respaldo como baterías.

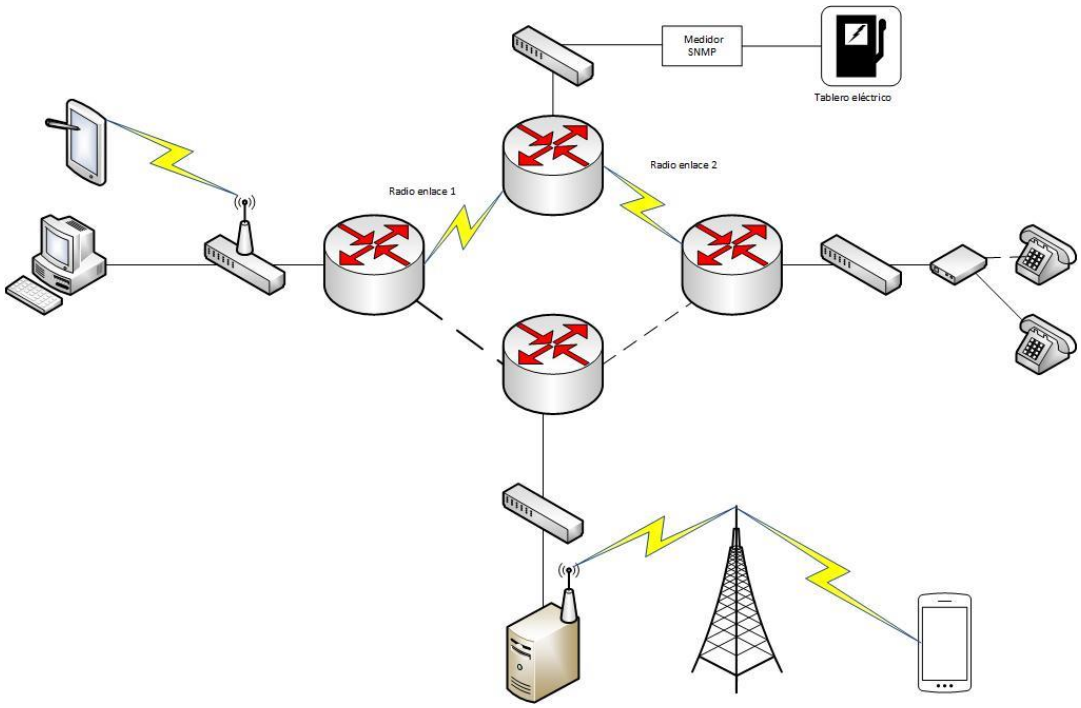


Figura 46: Red propuesta para pruebas de laboratorio.

Se busca ejemplificar equipo de telecomunicaciones y externo que se puedan encontrar en sitios de telecomunicaciones, por lo cual se tomara como base la red de la figura 46, tratando de ahondar en la configuración y preparación para el monitoreo de sus componentes como se detalla a continuación los elementos de a monitorear son:

3.1 ROUTER CISCO 2811

Figura 47: Router 2811

```
Router#configure terminal
Router(config)#snmp-server community public RO      ; habilitar como solo lectura la comunidad "public"
Router(config)#snmp-server community private RW     ;habilita como lectura y escritura la comunidad "private"
```

3.2 SWITCH CATALYST 3500

Al igual que los routers los Switchs que son dispositivos de capa 2, pueden transmitir información de manera nativa en el protocolo SNMP

Figura 48: Switch Catalyst 3500

La configuración del protocolo SNMP es igual a la desarrollada en el Router antes mencionado

3.3 MEDICIÓN DE VARIABLES EN EQUIPOS NO NATIVOS DEL PROTOCOLO SNMP

Haciendo uso del modulo Ethernet de Arduino y la librería AGentDuino, es posible transmitir las entradas análogas del Arduino y los sensores conectados a ellas, con lo cual es posible medir variables del entorno de la planta, como humedad relativa, temperatura, proximidad, detección de humo entre otras.

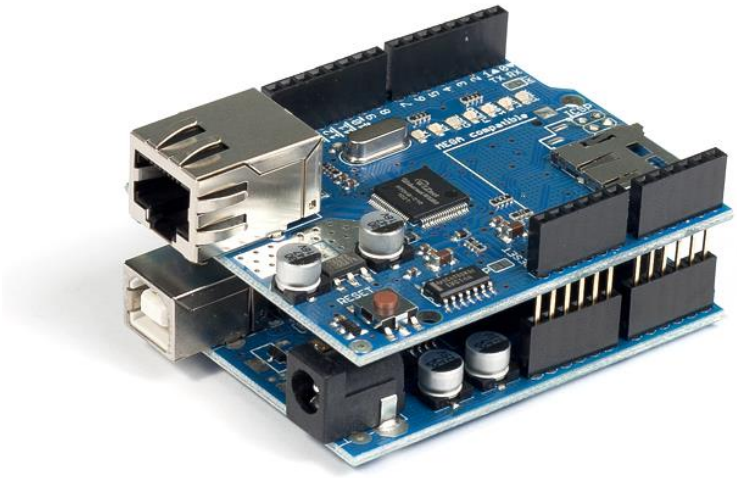


Figura 49: Arduino y modulo Ethernet.

El proyecto Agent Duino, es modificado y utilizado para la creación de las interfaces adaptativas, el código fuente modificada se presenta a continuación:

```
//librerias para publicacion SNMP
#include <Streaming.h>        // Include the Streaming library
#include <Ethernet.h>         // Include the Ethernet library
#include <SPI.h>
#include <MemoryFree.h>
#include <Agentuino.h>
#include <Flash.h>

//definiciones de analog read
int sensorPin = A0;    // select the input pin for the potentiometer
int ledPin = 13;      // select the pin for the LED
int sensorValue = 0;  // variable to store the value coming from the sensor

//..... Configurando para la correcta comunicacion en la red..
static byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
static byte ip[] = {192,168,1,177};
static byte gateway[] = {192,168,1,1};
static byte subnet[] = { 255, 255, 255, 0 };
```

```
//==========Se crea una cadena de caracteres que obtienen los OID
static char sysDescr[] PROGMEM      = "1.3.6.1.2.1.1.1.0";  // read-only  (DisplayString)
// .iso.org.dod.internet.mgmt.mib-2.system.sysObjectID (.1.3.6.1.2.1.1.2)
static char sysUpTime[] PROGMEM     = "1.3.6.1.2.1.1.3.0";  // read-only  (TimeTicks)
// .iso.org.dod.internet.mgmt.mib-2.system.sysContact (.1.3.6.1.2.1.1.4)
static char sysContact[] PROGMEM    = "1.3.6.1.2.1.1.4.0";  // read-write (DisplayString)
// .iso.org.dod.internet.mgmt.mib-2.system.sysName (.1.3.6.1.2.1.1.5)
static char sysName[] PROGMEM       = "1.3.6.1.2.1.1.5.0";  // read-write (DisplayString)
// .iso.org.dod.internet.mgmt.mib-2.system.sysLocation (.1.3.6.1.2.1.1.6)
static char sysLocation[] PROGMEM   = "1.3.6.1.2.1.1.6.0";  // read-write (DisplayString)
// .iso.org.dod.internet.mgmt.mib-2.system.sysServices (.1.3.6.1.2.1.1.7)
static char sysServices[] PROGMEM   = "1.3.6.1.2.1.1.7.0";  // read-only  (Integer)

//============Creando los OID para la variable a medir en este caso
// SysVoltage sera de voltaje
static char SysVoltage[] PROGMEM      = "1.3.6.1.3.1.3.2.1.2.0";  // read-only  (ObjectIdentifier)

// RFC1213 local values
static char locDescr[]            = "Agentuino, un agente SNMP ";           // read-only (static)
static uint32_t locUpTime         = 0;                                      // read-only (static)
static char locContact[20]        = "Univ.ElSalvador.UES";    // should be stored/read from EEPROM -
read/write (not done for simplicity)
static char locName[20]           = "Agentuino";              // should be stored/read from EEPROM -
read/write (not done for simplicity)
static char locLocation[20]       = "El Salvador";            // should be stored/read from EEPROM -
read/write (not done for simplicity)
static int32_t locServices        = 6;                                      // read-only (static)

// ===============Creando las variables de tipo int32_t para enviarla mediante SNMP
static int32_t VoltageObject         = 1;   //creando la variable tipo entero SNMP para temperatura
uint32_t prevMillis = millis();
char oid[SNMP_MAX_OID_LEN];
SNMP_API_STAT_CODES api_status;
SNMP_ERR_CODES status;

void pduReceived()
{
  SNMP_PDU pdu;
  api_status = Agentuino.requestPdu(&pdu);
  //
  if ((pdu.type == SNMP_PDU_GET || pdu.type == SNMP_PDU_GET_NEXT || pdu.type ==
SNMP_PDU_SET)
    && pdu.error == SNMP_ERR_NO_ERROR && api_status == SNMP_API_STAT_SUCCESS ) {
    //
    pdu.OID.toString(oid);
    // Implementation SNMP GET NEXT
    if ( pdu.type == SNMP_PDU_GET_NEXT ) {
```

```
char tmpOIDfs[SNMP_MAX_OID_LEN];
if ( strcmp_P( oid, sysDescr ) == 0 ) {
  strcpy_P ( oid, sysUpTime );
  strcpy_P ( tmpOIDfs, sysUpTime );
  pdu.OID.fromString(tmpOIDfs);
} else if ( strcmp_P(oid, sysUpTime ) == 0 ) {
  strcpy_P ( oid, sysContact );
  strcpy_P ( tmpOIDfs, sysContact );
  pdu.OID.fromString(tmpOIDfs);
} else if ( strcmp_P(oid, sysContact ) == 0 ) {
  strcpy_P ( oid, sysName );
  strcpy_P ( tmpOIDfs, sysName );
  pdu.OID.fromString(tmpOIDfs);
} else if ( strcmp_P(oid, sysName ) == 0 ) {
  strcpy_P ( oid, sysLocation );
  strcpy_P ( tmpOIDfs, sysLocation );
  pdu.OID.fromString(tmpOIDfs);
} else if ( strcmp_P(oid, sysLocation ) == 0 ) {
  strcpy_P ( oid, sysServices );
  strcpy_P ( tmpOIDfs, sysServices );
  pdu.OID.fromString(tmpOIDfs);
} else if ( strcmp_P(oid, sysServices ) == 0 ) {
  strcpy_P ( oid, "1.0" );
} else {
    int ilen = strlen(oid);
    if ( strncmp_P(oid, sysDescr, ilen ) == 0 ) {
      strcpy_P ( oid, sysDescr );
      strcpy_P ( tmpOIDfs, sysDescr );
      pdu.OID.fromString(tmpOIDfs);
    } else if ( strncmp_P(oid, sysUpTime, ilen ) == 0 ) {
      strcpy_P ( oid, sysUpTime );
      strcpy_P ( tmpOIDfs, sysUpTime );
      pdu.OID.fromString(tmpOIDfs);
    } else if ( strncmp_P(oid, sysContact, ilen ) == 0 ) {
      strcpy_P ( oid, sysContact );
      strcpy_P ( tmpOIDfs, sysContact );
      pdu.OID.fromString(tmpOIDfs);
    } else if ( strncmp_P(oid, sysName, ilen ) == 0 ) {
      strcpy_P ( oid, sysName );
      strcpy_P ( tmpOIDfs, sysName );
      pdu.OID.fromString(tmpOIDfs);
    } else if ( strncmp_P(oid, sysLocation, ilen ) == 0 ) {
      strcpy_P ( oid, sysLocation );
      strcpy_P ( tmpOIDfs, sysLocation );
      pdu.OID.fromString(tmpOIDfs);
      } else if ( strncmp_P(oid, sysServices, ilen ) == 0 ) {
      strcpy_P ( oid, sysServices );
      strcpy_P ( tmpOIDfs, sysServices );
```

```
      pdu.OID.fromString(tmpOIDfs);
      }
  }
}
// End of implementation SNMP GET NEXT / WALK

if ( strcmp_P(oid, sysDescr ) == 0 ) {
  // handle sysDescr (set/get) requests
  if ( pdu.type == SNMP_PDU_SET ) {
    // response packet from set-request - object is read-only
    pdu.type = SNMP_PDU_RESPONSE;
    pdu.error = SNMP_ERR_READ_ONLY;
  } else {
    // response packet from get-request - locDescr
    status = pdu.VALUE.encode(SNMP_SYNTAX_OCTETS, locDescr);
    pdu.type = SNMP_PDU_RESPONSE;
    pdu.error = status;
  }
  //
} else if ( strcmp_P(oid, sysUpTime ) == 0 ) {
  // handle sysName (set/get) requests
  if ( pdu.type == SNMP_PDU_SET ) {
    // response packet from set-request - object is read-only
    pdu.type = SNMP_PDU_RESPONSE;
    pdu.error = SNMP_ERR_READ_ONLY;
  } else {
    // response packet from get-request - locUpTime
    status = pdu.VALUE.encode(SNMP_SYNTAX_TIME_TICKS, locUpTime);
    pdu.type = SNMP_PDU_RESPONSE;
    pdu.error = status;
  }
  //
} else if ( strcmp_P(oid, sysName ) == 0 ) {
  // handle sysName (set/get) requests
  if ( pdu.type == SNMP_PDU_SET ) {
    // response packet from set-request - object is read/write
    status = pdu.VALUE.decode(locName, strlen(locName));
    pdu.type = SNMP_PDU_RESPONSE;
    pdu.error = status;
  } else {
    // response packet from get-request - locName
    status = pdu.VALUE.encode(SNMP_SYNTAX_OCTETS, locName);
    pdu.type = SNMP_PDU_RESPONSE;
    pdu.error = status;
  }
  //
} else if ( strcmp_P(oid, sysContact ) == 0 ) {
  // handle sysContact (set/get) requests
```

```
  if ( pdu.type == SNMP_PDU_SET ) {
    // response packet from set-request - object is read/write
    status = pdu.VALUE.decode(locContact, strlen(locContact));
    pdu.type = SNMP_PDU_RESPONSE;
    pdu.error = status;
  } else {
    // response packet from get-request - locContact
    status = pdu.VALUE.encode(SNMP_SYNTAX_OCTETS, locContact);
    pdu.type = SNMP_PDU_RESPONSE;
    pdu.error = status;
  }
  //
 } else if ( strcmp_P(oid, sysLocation ) == 0 ) {
  // handle sysLocation (set/get) requests
  if ( pdu.type == SNMP_PDU_SET ) {
    // response packet from set-request - object is read/write
    status = pdu.VALUE.decode(locLocation, strlen(locLocation));
    pdu.type = SNMP_PDU_RESPONSE;
    pdu.error = status;
  } else {
    // response packet from get-request - locLocation
    status = pdu.VALUE.encode(SNMP_SYNTAX_OCTETS, locLocation);
    pdu.type = SNMP_PDU_RESPONSE;
    pdu.error = status;
  }
  //
 } else if ( strcmp_P(oid, sysServices) == 0 ) {
  // handle sysServices (set/get) requests
  if ( pdu.type == SNMP_PDU_SET ) {
    // response packet from set-request - object is read-only
    pdu.type = SNMP_PDU_RESPONSE;
    pdu.error = SNMP_ERR_READ_ONLY;
  } else {
    // response packet from get-request - locServices
    status = pdu.VALUE.encode(SNMP_SYNTAX_INT, locServices);
    pdu.type = SNMP_PDU_RESPONSE;
    pdu.error = status;
  }

 //======== INICIO DE PETICION Y ENVIO DE VALORES POR SNMP======
 //======== PETICION Y ENVIO DEL VALOR DE VOLTAJE ===========
    } else if ( strcmp_P(oid, SysVoltage) == 0 ) {
  // handle  (set/get) requests
  if ( pdu.type == SNMP_PDU_SET ) {
    // response packet from set-request - object is read-only
    pdu.type = SNMP_PDU_RESPONSE;
    pdu.error = SNMP_ERR_READ_ONLY;
  } else {
```

```
      // se responde con un paquete que contiene el valro de la temperatura TEmpObjet
      status = pdu.VALUE.encode(SNMP_SYNTAX_INT, VoltageObject);
      pdu.type = SNMP_PDU_RESPONSE;
      pdu.error = status;
     }


   //FIN  DE ENVIO DE LOS VALORES SENSADOS.....

   } else {
     // oid does not exist
     // response packet - object not found
     pdu.type = SNMP_PDU_RESPONSE;
     pdu.error = SNMP_ERR_NO_SUCH_NAME;
   }
   //
   Agentuino.responsePdu(&pdu);
  }
  //
  Agentuino.freePdu(&pdu);
  //
}

Void setup()
{
   Serial.begin(9600);
   Ethernet.begin(mac, ip);

  //Inicando las funciones de Agentuino
  api_status = Agentuino.begin();

  if ( api_status == SNMP_API_STAT_SUCCESS ) {
   //
   Agentuino.onPduReceive(pduReceived);
   //
   delay(10);
   //
   return;
  }
  //
  delay(10);
}

Void loop() {
  delay (1000);
  sensorValue = (10*analogRead(sensorPin)/512);
  Serial.println(sensorValue);
  //programa para leer estado inicio
```

```
 //digitalWrite(ledPin, HIGH);
 //delay(sensorValue);
 //digitalWrite(ledPin, LOW);
 //delay(sensorValue);
 //programa para leer estado fin

//Se igualan los datos obtenidos con las variables que seran enviada por SNMP
 VoltageObject  = sensorValue; // valores de Temperatura
// listen/handle for incoming SNMP requests
 Agentuino.listen();

 if ( millis() - prevMillis > 1000 ) {
   // increment previous milliseconds
   prevMillis += 1000;
   //
   // increment up-time counter
   locUpTime += 100;
 }

}
```

Con lo cual podemos censar cualquiera de las entradas análogas de la tarjeta Arduino, siendo posible la utilización de sensores de temperatura, humo, iluminación, humedad relativa. Etc.

3.3.1 USO DE TARJETA FLEX Q3

El modulo de gestión del fabricante permite la configuración de la tarjeta de manera rápida y eficiente, estableciendo la comunicación de la misma mediante el protocolo HTTP o SSH, se debe establecer la conexión mediante la IP asociada al dispositivo, por defecto se cuenta con la IP: 192.168.1.10, y la clave de acceso es un espacio en blanco, luego de solicitada la conexión es posible habilitar las opciones de configuración, el menú principal nos permite cambiar la IP de acceso y las credenciales de red, como la máscara de sub red o el Gateway, además de los puertos de acceso al servidor de la tarjeta.

Acceso a la tarjeta

Para acceder a la tarjeta y su gestión es necesario conectar la tarjeta por el puerto Ethernet a una red en la cual sea accesible, la configuración por defecto de la tarjeta requiere conectarla en una red de la familia 192.167.1.0, valor que se puede cambiar luego de la configuración inicial a una red más conveniente

Debemos entonces seleccionar la IP por defecto de la tarjeta la cual es 192.167.1.10 con lo cual se habilitaran las configuraciones de la tarjeta como se observa en le figura 51.

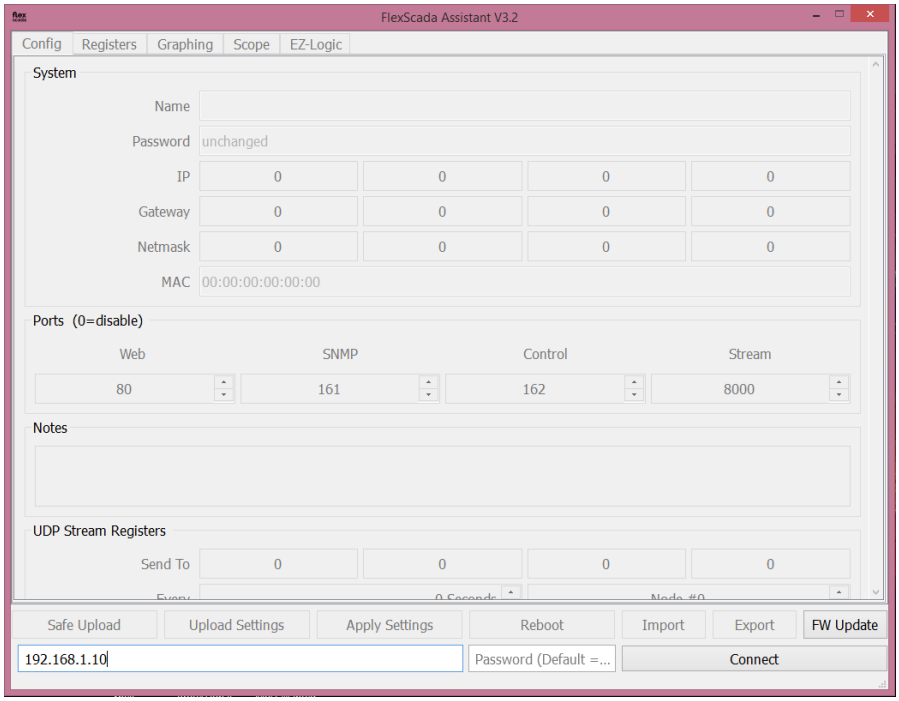


Figura 50: Programa de gestión y configuración Flex Q3

Una vez habilitada la tarjeta es posible cambiar los valores de IP, Gateway, Mascara de sub red o MAC, además del nombre asociado a la tarjeta y el password para accesar a ella.

Ademas de los puertos habilitados para las conexiones WEB, SNMP, control y stream.

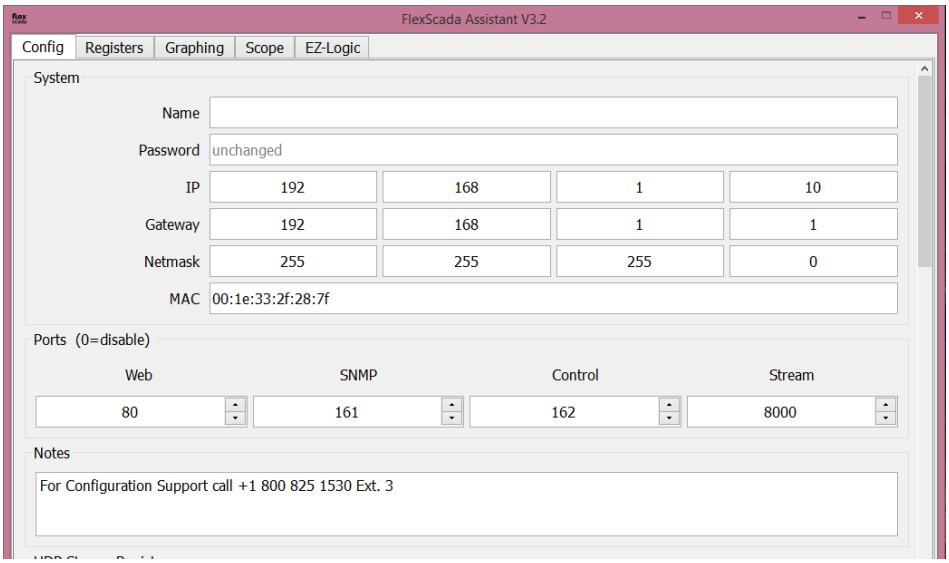


Figura 51: Configuración de red de tarjeta Flex Q3

Configuración de entradas Analógicas.

A continuación se presenta la configuración básica de las entradas análogas de la tarjeta, siendo las entradas #1 y #2 configuradas para la medición de voltaje DC, específicamente una batería de UPS, se calibran las entradas con una ganancia unitaria para no afectar la medición, las entradas #3, #4 y #5, están destinadas a la medición trifásica de un tablero eléctrico, los valores de multiplicador y offset se calculan a partir de la resistencia del divisor de tensión seleccionada, este tema se retomara con mayor énfasis en el capítulo tres del presente documento, finalmente las entradas #6, #7 y #8 corresponden la medición de corriente de un tablero eléctrico, los valores de multiplicación y offset son calculados a partir del sensor de corriente seleccionado para la medición.

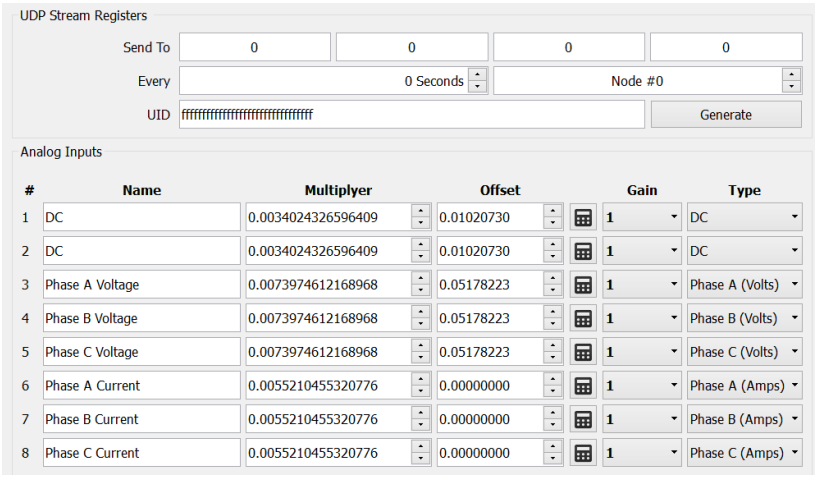


Figura 52: Configuración de entradas análogas de tarjeta

El esquema de cálculo de para los valores de ganancia, desfase y multiplicador se realiza mediante la opción de cálculo presente en la pantalla principal, a partir de los valores de las resistencias del divisor de tensión, los transformadores de corriente seleccionados o los niveles pico de voltaje y corriente a medir

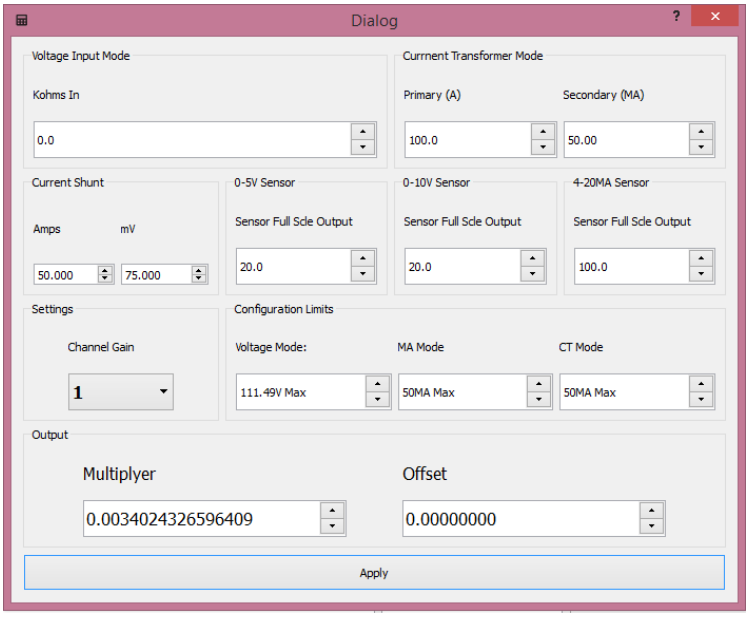


Figura 53: Configuración de canales de adquisición de datos.

La configuración de los canales de adquisición de datos nos permite configurar la resolución, referencia y muestreo de los ADC internos, además de la frecuencia de muestro del canal y la frecuencia de cruce por cero.

Figura 54: Configuración de las salidas

Podemos configurar las 8 salidas de estado sólido de la tarjeta, asignándoles un estado inicial y monitoreando el estado actual, estos estados varían a partir de las comparaciones lógicas que se realicen tomando las entradas como operadores.

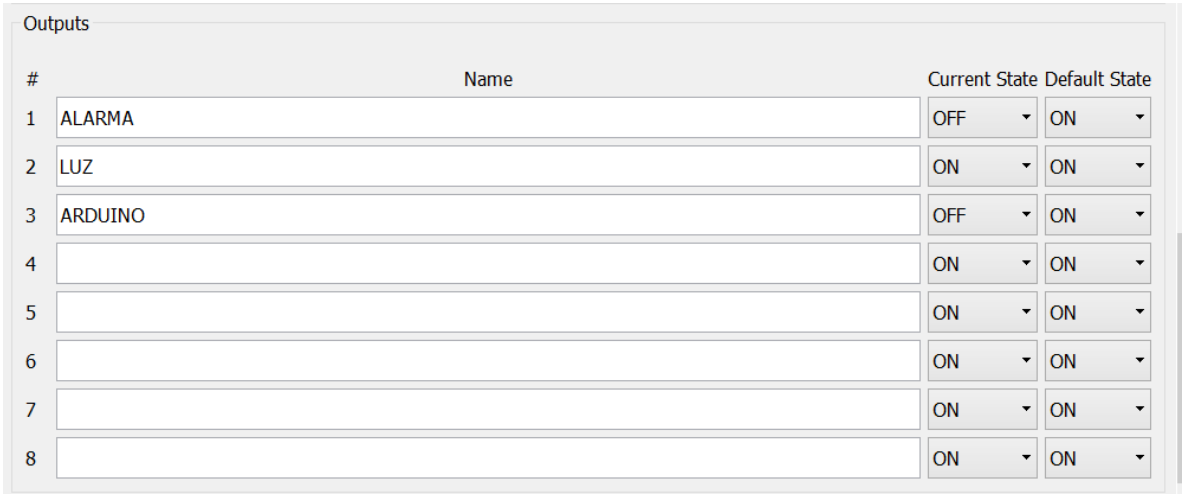


Figura 55: Configuración de salidas

Configuración de registros

En este apartado de la segunda pestaña del programa de configuración, podemos acceder a los valores instantáneos de los diferentes registros de la tarjeta, estos registros incluyen las entradas y salidas de la tarjeta.

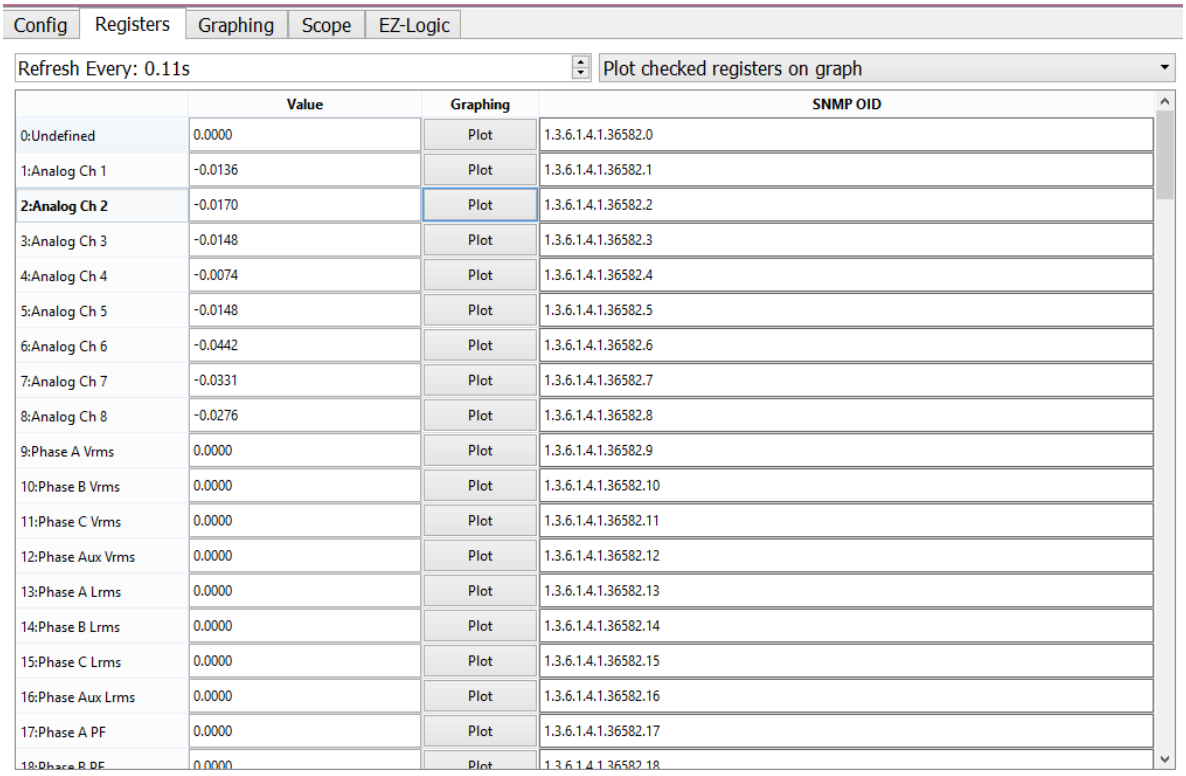


Figura 56: Monitoreo de valores instantáneos de registros

Como se observa en la figura 56, es posible monitorear los valores instantáneos de las entradas de la tarjeta mediante la pestaña de registros, como se observa también podemos obtener de esta pestaña las OID de cada entrada para la configuración del software de monitoreo.

Seleccionando la opción plot que acompaña a cada registro, en la pestaña Graphing, se genera una gráfica de los valores instantáneos de la entrada seleccionada como se ve en la figura 57.

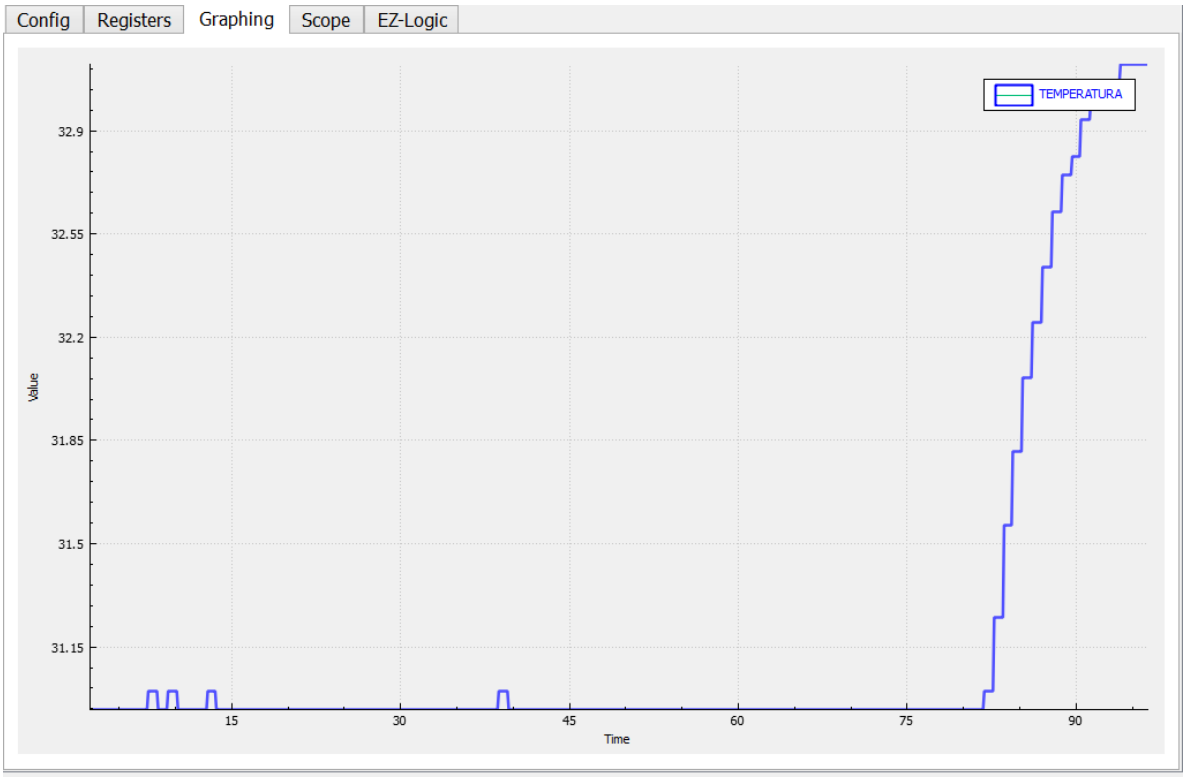


Figura 57: Grafico de valores de registros instantáneos

4. CONCLUSIONES

- La implementación de interfaces adaptativas para elementos complementarios de los sistemas de comunicación y la correcta gestión y monitoreo de los mismos, aumenta de manera significativa la confiabilidad del sistema y garantiza la disponibilidad, pues se atacan preventivamente variables externas que de otra manera podrían afectar significativamente la operación normal de los equipos de telecomunicaciones.

  Es importante garantizar el funcionamiento adecuado de los elementos que componen los sistemas de telecomunicaciones, como la temperatura ambiente, o la calidad de la energía utilizada para el funcionamiento del sitio de telecomunicaciones.

- La importancia de la implementación de mediciones en equipos como UPS, Aires acondicionados, Plantas de generación auxiliar o bancos de baterías, radica en lograr mantener un servicio constante y adecuado mediante la prevención de eventos fuera de la cotidianidad de la operación de los equipos de telecomunicaciones, como fallas de suministro eléctrico o fenómenos atmosféricos que afecten su funcionamiento cotidiano.

  Equipos que permitan este tipo de monitoreo, resultan en un aumento significativo del costo de los sistemas y en casos de sistemas ya implementados a menudo se cuenta con equipos que no cuentan con la capacidad de implementación de monitoreo SNMP.

- Se presentaron dos soluciones para la generación de interfaces adaptativas, de las cuales, la tarjeta Arduino representa una opción económica cuando no es necesario tener una precisión significativa en las mediciones, por ejemplo, el estado de una planta de generación auxiliar o la medición de variables ambientales como temperatura.

  Pero representa una desventaja a causa de la baja velocidad de adquisición de datos o la resolución del convertidor, al ser comparada con la tarjeta FlexQ3, sin embargo el sistema pueden mejorarse de manera significativa si se integra un chip como el ADE7753, capaz de medir variables eléctricas como voltaje, corriente, factor de potencia, y potencias real, reactiva y aparente monofásicas, con los cual se pueden implementar sistemas aislados que aumenten la precisión de las mediciones quitando la medición de las variables a la placa Arduino.

- El uso de la tarjeta SNMP Flex Q3, facilita la medición de voltaje AC, DC al igual que las respectivas corrientes, al igual que la utilización de sensores OneWire mediante el puerto correspondiente, teniendo una velocidad y resolución de adquisición de datos muy precisos, sin embargo, a menudo es necesario la generación de indicadores como SAIFI y SAIDI o el cálculo de variables como factor de potencia, por lo tanto según se revisó en los diferentes software se podría utilizar la base de datos de PANDORA para generar consultas en MySql que permitan el cálculo de indicadores o adecuación de la información.

- El software de monitoreo CACTI presenta una versatilidad considerable gracias a la presentación plana de sus gráficos y el uso adecuado de la herramienta RdTools para el mejor uso del almacenamiento de la base de datos, por lo cual es una solución útil en sistemas donde ya existe un servidor dedicado a otras funciones y de desea incluir una herramienta de monitoreo, como se revisó en el capítulo II es capaz de ser instalado inclusive en una tarjeta Raspberry pi.

- Finalmente se consideró en el capítulo II que Nagios representa la mejor opción para el monitoreo de los sistemas de telecomunicación y las variables adaptativas asociadas a ellos, pues la posibilidad de utilizar un asistente para la configuración de equipos como Routers o Switchs facilita su monitoreo, además de ser amigable con MIB generadas a partir de interfaces adaptativas creadas por tercero y que contienen información que no es nativa del protocolo SNMP.

5. BIBLIOGRAFÍA

Request for Comments

- RFC 1157 (SNMP, 1990)
- RFC 3410 (SNMPv3, 2002)

Internet

- http://www.cisco.com/c/en/us/products/collateral/switches/catalyst-2960-series-switches/product_data_sheet0900aecd80322c0c.pdf
- http://www.cacti.net/spine_info.php
- https://www.nagios.com/products/nagios-xi/
- http://sourceforge.net/projects/pandora/files/Pandora%20FMS%205.1/Final/Documentation/PandoraFMS_5.1_Manual_ES.pdf/download
- https://www.es.paessler.com/prtg/product-information
- https://code.google.com/p/agentuino/

Libros

- Redes Cisco CCNP a Fondo. Guía de estudio para profesionales, Ernesto Ariganello y Enrique Barrientos Sevilla, Madrid, España