Congr. Int. en Ing. Electrónica. Mem. ELECTRO, Vol. 39, pp. 160-165, Oct 2017, Chihuahua, Chih. México
http://electro.itchihuahua.edu.mx/memorias_electro/MemoriaElectro2017.zip
ISSN 1405-2172

NAGIOS, COMO HERRAMIENTA DE MONITOREO DE REDES DE COMPUTADORAS

José Francisco Alatorre-Avila[1], José Antonio Hoyo-Montaño, David Sáenz-Zamarrón, Nancy Ivette Arana-de-las-Casas, Enrique García-Grajeda, Miriam Aide Mendoza Mendoza.

[1]TECNM/Instituto Tecnológico de Cd. Cuauhtémoc
Departamento de Sistemas y Computación
Av. Tecnológico No. 137, Cuauhtémoc, Chih. México C.P. 31500
01 52 (625) 581 1707.
jalatorre@itcdcuauhtemoc.edu.mx

RESUMEN

El requerimiento de servicios de cómputo confiables, en todo momento, es, hoy en día imprescindible. En este trabajo se describe un esquema de monitoreo de los recursos en una red de computadoras, este ofrece alertas para que un administrador realice las acciones correctivas pertinentes. Se instaló en tres computadoras con las mismas características las distribuciones Linux: Centos, Fedora y Ubuntu para comparar el comportamiento de cada una y decidir cuál de ellas es la más estable para utilizarla con la herramienta de monitoreo. Los elementos de monitoreo se configuraron en NAGIOS, estableciendo varios parámetros, antes de notificar, así como también el procedimiento para enviar las alertas SMS con GAMMU. Finalmente, se analizaron los resultados de las pruebas realizadas y la información obtenida en la instalación de las distribuciones LINUX, se obtuvo una configuración adecuada de alertas de NAGIOS en CENTOS.

Palabras clave: Nagios, Centos, Gammu.

ABSTRACT

The requirement of reliable computing services, at all times, is nowadays essential. This paper describes an outline of resources monitoring in a computer network, offering alerts so that an administrator performs the relevant corrective actions. Linux distributions: Centos, Fedora and Ubuntu were installed on three computers with the same characteristic to compare the behavior of each one and decide which of them is the most stable for use with the monitoring tool. The elements of monitoring were configured in NAGIOS, establishing various parameters, before notification, as well as the procedure to send SMS alerts with GAMMU. Finally, the results of the tests performed and information obtained from the installation of LINUX distributions was analyzed, obtaining a proper configuration of NAGIOS alerts on CENTOS.

Keywords: Nagios, Centos, Gammu.

1. INTRODUCCIÓN

Las redes de telecomunicaciones siguen creciendo significativamente, muchos servicios y aplicaciones de gran importancia, se ejecutan en redes de baja confiabilidad. Un fallo de uno de los servicios puede resultar en gran pérdida para una empresa, como por ejemplo, en una institución educativa donde las inscripciones de los estudiantes se realizan por Internet y el servidor donde se encuentra la aplicación de inscripciones, se encuentre fuera de servicio, cada momento, produciría un caos en los estudiantes y se perdería la credibilidad del proceso, en un ejemplo como este se manifiesta la gran importancia que los servicios de red estén funcionando el mayor tiempo posible, claro con la existencia de la posibilidad de fallo, pero que esta sea mínima[1].

NAGIOS es un Sistema Open Source para monitorizar la red, sus servicios y los recursos de los hosts, informa de la disponibilidad y de las posibles causas de errores. IT Central Station en su reporte de Mayo de 2017 de Software de Monitoreo de Redes [3], realiza un análisis de cuáles son los 10 mejores, estando como número uno NAGIOS.

Se puede acceder a él mediante una interfaz web donde se puede visualizar todo lo monitorizado para identificar las posibles fallas y lo que está funcionando adecuadamente, maneja un código de colores, donde: verde es correcto, amarillo algo no va del todo bien, rojo es estado crítico. Existe un sistema de alarmas que alerta al administrador de la red de este fallo enviando e-mail, mensajes de texto, o ambos. Cada una de estas actividades puede ser configurada modificando los archivos, según las necesidades del administrador [2]. Para este trabajo se utilizó la herramienta GAMMU para enviar mensajes de texto SMS por medio de un celular conectado al equipo de cómputo de pruebas. Así mismo está herramienta Open Source se instaló en tres distribuciones de Linux para realizar una comparación y obtener información que permita determinar cual de ellas es la mejor para utilizarse con NAGIOS.

2. IMPLEMENTACIÓN DE NAGIOS

NAGIOS es un sistema de monitoreo de redes de codigo abierto, para vigilar el hardware y servicios que se especifiquen, alertando cuando el comportamiento de los mismos no sea el deseado. Para este trabajo se utilizó la versión 3.2.3 para UBUNTU, y la versión 3.5.1 para CENTOS y FEDORA.

2.1. Metodología

La detección oportuna de fallas y el monitoreo son actividades de gran relevancia para brindar un buen servicio de cómputo que se realizan, ya sea con un enfoque activo o pasivo [4].

Monitoreo Activo. Se realiza inyectando paquetes de prueba en la red, cada determinado tiempo.

Monitoreo Pasivo. Se basa en la obtención de datos a partir de recolectar y analizar el tráfico que circula por la red. Utiliza

SNMP (Protocolo Simple de Administración de Red) para recolectar información.
El monitoreo realizado es Activo y Pasivo, considerando lo siguiente: se consideraron 2 divisiones para analizar los dispositivos, una desde la perspectiva de los dispositivos de red o hosts, y otra desde los servicios que proporcionan. De esta manera los servicios y dispositivos son los siguientes:
Servicios: HTTP, SSH, PING.
Dispositivos: Servidor WEB, Servidor de Monitoreo, Router CISCO 2811, Switch CISCO catalyst 2950, Punto de Acceso.

2.2. Distribuciones de Linux

Para este trabajo se utilizaron tres equipos de cómputo con las mismas características cada uno, las cuales fueron las siguientes:
Procesador. Pentium 4 de 3.0 GHz DUAL CORE.
RAM. 768 MB y 8 MB de memoria de video compartida.
Disco Duro. 80 GB.
Tarjeta Madre. ASROCK 775i65G.
Tarjeta de Red. 10/100 integrada.
Las distribuciones de Linux utilizadas fueron: Ubuntu 12.04, Fedora 18, Centos 6.6.

2.3. Protocolo de Monitoreo

El monitoreo se realizó cada 5 minutos para hosts y cada 10 minutos para servicios, utilizando el agente propio de NAGIOS y/o el protocolo SNMP. Basados en ese contexto se procedió a definir la Tabla 1, con los parámetros de prueba que se van a monitorear.

Tabla 1. Componentes de Monitoreo

HOST	TIPO	SERVICIO
C2811	ROUTER	TIEMPO DE ACTIVIDAD
		CONECTIVIDAD
		ANCHO DE BANDA INTERFACES
C2950	SWITCH	TIEMPO DE ACTIVIDAD
		CONECTIVIDAD
		ANCHO DE BANDA INTERFACES
UBUNTU	SERVIDOR	SERVIDOR WEB
		ADMINISTRACIÓN REMOTA
		CONECTIVIDAD
FEDORA	SERVIDOR	SERVIDOR WEB
		ADMINISTRACIÓN REMOTA
		CONECTIVIDAD
CENTOS	SERVIDOR	SERVIDOR WEB
		ADMINISTRACION REMOTA
		SERVIDOR WEB
LOCALHOST	SERVIDOR	USUARIOS ACTIVOS
		PROCESOS EN EJEUCIÓN
		CARGA DEL PROCESADOR

La configuración del servidor de monitoreo con la herramienta NAGIOS, se realiza a través de los archivos, como se muestra en la Tabla 2.

Tabla 2. Archivos de Configuración NAGIOS

ARCHIVO	DESCRIPCION	RUTA
nagios.cfg	Archivo de configuración principal, donde se establecen las rutas que van a tener los archivos para el funcionamiento de NAGIOS.	/etc/nagios/
commands.cfg	Contiene los comandos predefinidos para el usuario.	/etc/nagios/ objects
contacts.cfg	Contiene información de los usuarios a quien se le van enviar las alertas.	/etc/nagios/ objects
timeperiods.cfg	Contiene información sobre los periodos de notificaciones.	/etc/nagios/ objects
templates.cfg	Contiene información de plantillas genéricas para hosts y servicios.	/etc/nagios/ objects
localhost.cfg	Contiene información sobre el equipo donde está instalado NAGIOS.	/etc/nagios/ objects
switch.cfg	Contiene información de los routers y switches.	/etc/nagios/ objects
server.cfg	Contiene información de los servicios a monitorear de los servidores.	/etc/nagios/ objects

Cabe mencionar que en UBUNTU la ruta de los archivos de configuración como *nagios.cfg*, se encuentra en */etc/nagios3* y los demás están en */etc/nagios3/conf.d/*.

2.4. Definición de Hosts y Servicios

En la Tabla 3 se definen las instrucciones necesarias para configurar los hosts en cada uno de los archivos correspondientes.

Tabla 3. Definición de HOSTS NAGIOS

HOST	ARCHIVO	COMANDO

C2811	switch.cfg	define host{
		use generic-router
		host_name C2811
		alias C2811
		address 10.10.12.254
		parents C2950
		hostgroups routers
		}

C2950	switch.cfg	define host{
		use generic-switch
		host_name C2950
		alias C2950
		address 10.10.12.254
		hostgroups switches
		}

UBUNTU	server.cfg	define host{
		use linux-server
		host_name NagiosU
		parents C2950
		alias UBUNTU
		address 10.10.12.178
		}

FEDORA	server.cfg	define host{
		use linux-server
		host_name nagiosf
		parents C2950
		alias FEDORA
		address 10.10.12.236
		}

CENTOS	server.cfg	define host{
		use linux-server
		host_name nagiosc
		parents C2950
		alias CENTOS
		address 10.10.12.58
		}

LOCALHOST	localhost.cfg	define host{
		use linux-server
		host_name localhost
		alias localhost
		parents C2950
		address 127.0.0.1
		}

La definición de las instrucciones necesarias para configurar servicios en cada archivo se realiza de la siguiente manera:

SERVICIO: Tiempo de Actividad
HOST: C2811
ARCHIVO: switch.cfg

```
define service{
 use                    generic-service
 host_name              C2811
 service_description    ACTIVO
 check_command check_snmp!-C monitor  -o sysUpTime.0
        }
```

SERVICIO: Conectividad
HOST: C2811
ARCHIVO: switch.cfg

```
define service{
        use                     generic-service
        host_name               C2811
        service_description     PING
        check_command  check_ping!200.0,20%!600.0,60%
        normal_check_interval   5
        retry_check_interval    1
        }
```

SERVICIO: Ancho de Banda Interfaces
HOST: C2811
ARCHIVO: switch.cfg

```
define service{
        use                     generic-service
        host_name               C2811
        service_description     FA0/0 ESTATUS
        check_command   check_snmp!-C  monitor  -o
ifOperStatus.2 -r 1 -m RFC1213-MIB
        }
define service{
        use                     generic-service
        host_name               C2811
        service_description     FA0/0 ANCHO DE BANDA
        check_command
        check_local_mrtgtraf!/var/www/mrtg/c2811/10.10.1
2.254_2.log!AVG!1000000,1000000!5000000,5000000!10
        }
```

SERVICIO: SERVIDOR WEB
HOST: CENTOS
ARCHIVO: server.cfg

```
define service{
      use                     local-service
      host_name               nagiosc
      service_description     HTTP
      check_command           check_http
}
```

SERVICIO: Administración Remota
HOST: CENTOS
ARCHIVO: server.cfg

```
define service{
      use                     generic-service
      host_name               nagiosc
      service_description     SSH
      check_command           check_ssh
}
```

SERVICIO: Usuarios Activos
HOST: LOCALHOST
ARCHIVO: localhost.cfg

```
define service{
      use                     local-service
      host_name               localhost
      service_description     Current Users
      check_command           check_local_users!20!50
      }
```

SERVICIO: Procesos en Ejecución
HOST: LOCALHOST
ARCHIVO: localhost.cfg

```
define service{
      use                     local-service
      host_name               localhost
      service_description     Total Processes
      check_command
      check_local_procs!250!400!RSZDT
      }
```

SERVICIO: Carga del Procesador
HOST: LOCALHOST
ARCHIVO: localhost.cfg

```
define service{
      use                     local-service
      host_name               localhost
      service_description     Current Load
      check_command
      check_local_load!1.4,1.2,1!1.8,1.6,1.4
      }
```

2.5. Ejecución del Monitoreo

El Protocolo de Monitoreo requirió que se establecieran parámetros como: tiempo de monitoreo, estados de servicios, intervalo de notificaciones e intentos de verificación antes de notificar. Estos se establecieron en la Tabla 4 de la siguiente manera.

Tabla 4. Parámetros de monitoreo

PARAMETRO	VALOR	DESCRIPCIÓN
Tiempo de monitoreo	5	Intervalo de tiempo que indica cada cuantos minutos se van a revisar los HOSTS.

	10	Intervalo de tiempo que indica cada cuantos minutos se van a revisar los servicios.
Estados de Monitoreo	DOWN	Estado en el cual el servicio/host esta desactivado o no se encuentra.
	WARNING	Es una advertencia que indica si el servicio se encuentra cerca del estado crítico.
	UNKNOWN	Indica que el servicio no se puede detectar.
	CRITICAL	Indica que el servicio se encuentra en un estado crítico.
	UP	Indica que el servicio/host están activos y funcionando.
	UNREACHABLE	Indica que el host no se puede detectar.
Intervalo de Notificaciones	60	Intervalo de tiempo que indica cada cuantos minutos se van enviar notificaciones del estado de los servicios, en el caso de que ocurra un cambio de estado.
	30	Intervalo de tiempo que indica cada cuantos minutos se van enviar notificaciones del estado de los hosts, en el caso de que ocurra un cambio de estado.

3. CONEXIÓN CON CELULAR

La mayoría de los teléfonos celulares existentes, permiten la conexión por medio de USB a la computadora, tienen su propio software de conexión diseñado por el fabricante del teléfono o por otra compañía, algunos son gratuitos y otros tienen costo, pero la gran mayoría están diseñados para plataformas Windows y la desventaja a parte del costo radica en la falta de manipulación del código fuente [5].

3.1. GAMMU

Existen diferentes herramientas de software libre que permiten la conexión y comunicación con la computadora a través del puerto USB, puerto serial, puerto infrarrojo y bluetooth. El más recomendado para establecer la comunicación y que además permite el desarrollar scripts para su utilización, es el GAMMU [6], el cual fue diseñado primeramente para celulares Nokia, pero ya tiene soporte para diferentes marcas, entre ellas SAMSUNG, ALCATEL, ZTE.
En al Figura 1 se muestra el resultado de ejecutar el comando *gammu --identify*

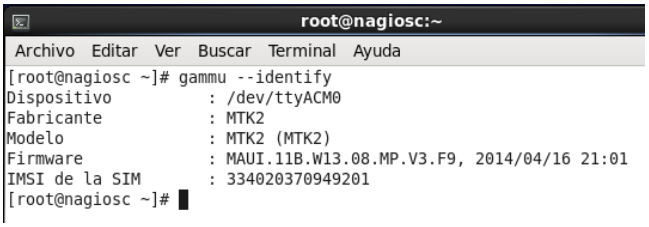


Fig 1. Identificación de celular con gammu Centos

3.2. Alertas y Notificaciones

Para configurar los parámetros de alertas y notificaciones se editó el archivo *templates.cfg* en cada una de las distribuciones de LINUX [7], se remarcó en negritas los parámetros importantes que se definieron en la Tabla 4, quedando de la siguiente manera

define host{
name linux-server
use generic-host
check_period 24x7
check_interval 5; ***Tiempo de monitoreo***
retry_interval 1
max_check_attempts 10
check_command check-host-alive
notification_period 24x7
notification_interval 30; Intervalo de notificaciones
notification_options d,u,r; ***Notificar estados DOWN, UNREACHABLE***
contact_groups admins
register 0
}
define service{
name generic-service
active_checks_enabled 1
passive_checks_enabled 1
obsess_over_service 1
check_freshness 0
notifications_enabled 1
event_handler_enabled 1
flap_detection_enabled 1
failure_prediction_enabled 1
check_period 24x7
max_check_attempts 3
normal_check_interval 10
retry_check_interval 2
contact_groups admins
notification_options w,u,c,r; ***Notificar estados WARNING, CRITICAL, UNREACHABLE, RECUPERACION***
notification_interval 60; ***Intervalo de notificaciones***
notification_period 24x7
register 0
}

Para configurar las alertas por medio de SMS [8], se modificó el archivo *commands.cfg* agregando comandos para enviar alertas por medio de gammu cuando cambie de estado un servicio o un HOST, identificando en el mensaje de alerta la distribución de LINUX que envía el mensaje. El comando resultante se muestra a continuación:

define command {
command_name notify-host-by-sms
command_line /usr/bin/printf "%b" "FEDORA: Alerta $HOSTNAME$ esta $HOSTSTATE$ F/H: $SHORTDATETIME$" | gammu sendsms TEXT $CONTACTPAGER$

```
}
define command {
 command_name notify-service-by-sms
 command_line /usr/bin/printf "%b" "FEDORA: Alerta servicio $SERVICEDESC$ en $HOSTNAME$ esta $SERVICESTATE$ F/H: $SHORTDATETIME$ " | gammu sendsms TEXT $CONTACTPAGER$
}
define command {
 command_name notify-host-by-sms
 command_line /usr/bin/printf "%b" "UBUNTU: Alerta $HOSTNAME$ esta $HOSTSTATE$ F/H: $SHORTDATETIME$" | gammu sendsms TEXT $CONTACTPAGER$
}
define command {
 command_name notify-service-by-sms
 command_line /usr/bin/printf "%b" "UBUNTU: Alerta servicio $SERVICEDESC$ en $HOSTNAME$ esta $SERVICESTATE$ F/H: $SHORTDATETIME$ " | gammu sendsms TEXT $CONTACTPAGER$
}
```

4. RESULTADOS

A continuación se muestran unas Tablas comparativas del rendimiento que tuvo cada uno de los equipos de cómputo, desde la perspectiva de la carga y uso del procesador, como también de la memoria RAM. Las Tablas 5, 6, 7 permiten identificar cual distribución de Linux es la que trabaja mejor, o dicho de otra manera cual es la que hace uso más eficiente de los recursos.

En la Tabla 5 se muestra la comparación de la Carga del CPU durante 1 minuto, 5 minutos, y 15 minutos, y se puede verificar que la distribución que mejor administra esta Carga es CENTOS, ya que las cantidades están establecidas en porcentaje, donde 1.0 sería el 100%, y tener una carga arriba del 50% ó .5 es preocupante.

Tabla 5. Carga de CPU en Equipos de Cómputo

	CARGA CPU		
	1 MIN	**5 MIN**	**15 MIN**
FEDORA	0.5	0.44	0.42
UBUNTU	1.24	0.88	0.69
CENTOS	**0.31**	**0.35**	**0.28**

La Tabla 6 de Uso de CPU describe que porcentaje del procesador está siendo usado, ya sea por el usuario, sistema, porcentaje libre y en espera. En ella se puede apreciar que CENTOS gestiona mejor el Uso del CPU.

Tabla 6. Uso de CPU en Equipos de Cómputo

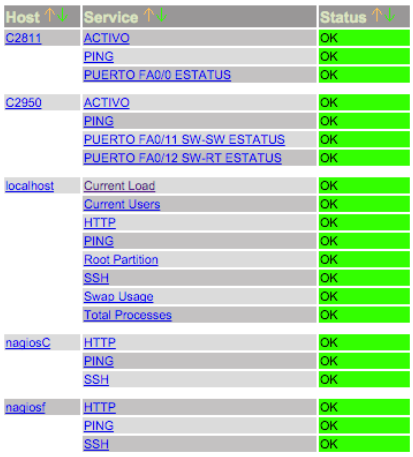
	USO CPU			
	USUARIO	**SISTEMA**	**LIBRE**	**ESPERA**
FEDORA	9.5	1.5	86.3	2.2
UBUNTU	16.1	3.3	72.5	7.9
CENTOS	**0.5**	**0.5**	**93.9**	**5.1**

En la Tabla 7 se realizó una comparación de cual distribución de Linux gestionaba mejor la Memoria RAM, y generando como resultado que CENTOS, tiene más memoria disponible para trabajar.

Tabla 7. Memoria RAM en Equipos de Cómputo

	MEMORIA RAM						
	TOTAL	**USADA**	**LIBRE**	**BUFFER**	**CACHE**	**LP**	**UP**
FEDORA	758664	666656	92008	10356	165008	267372	491292
UBUNTU	757076	704488	52588	6656	227128	286372	470704
CENTOS	759244	729892	29352	146788	157564	**333704**	425540
	LP	MEMORIA RAM DISPONIBLE					
	UP	MEMORIA UTILIZADA POR PROGRAMAS					

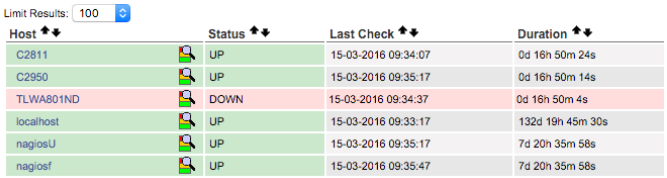
En la Figura 2, se muestra el resultado de la detección de los servicios configurados en NAGIOS.



Host	Service	Status
C2811	ACTIVO	OK
	PING	OK
	PUERTO FA0/0 ESTATUS	OK
C2950	ACTIVO	OK
	PING	OK
	PUERTO FA0/11 SW-SW ESTATUS	OK
	PUERTO FA0/12 SW-RT ESTATUS	OK
localhost	Current Load	OK
	Current Users	OK
	HTTP	OK
	PING	OK
	Root Partition	OK
	SSH	OK
	Swap Usage	OK
	Total Processes	OK
nagiosC	HTTP	OK
	PING	OK
	SSH	OK
nagiosf	HTTP	OK
	PING	OK
	SSH	OK

Fig. 2. Detección de servicios NAGIOS

La Figura 3 representa el resultado de la detección de los Hosts configurados en NAGIOS.



Limit Results: 100

Host	Status	Last Check	Duration
C2811	UP	15-03-2016 09:34:07	0d 16h 50m 24s
C2950	UP	15-03-2016 09:35:17	0d 16h 50m 14s
TLWA801ND	DOWN	15-03-2016 09:34:37	0d 16h 50m 4s
localhost	UP	15-03-2016 09:33:17	132d 19h 45m 30s
nagiosU	UP	15-03-2016 09:35:17	7d 20h 35m 58s
nagiosf	UP	15-03-2016 09:35:47	7d 20h 35m 58s

Fig. 3. Detección de Hosts NAGIOS

En la Figura 4 se muestra un ejemplo de un mensaje SMS enviado al momento de detectar un problema.

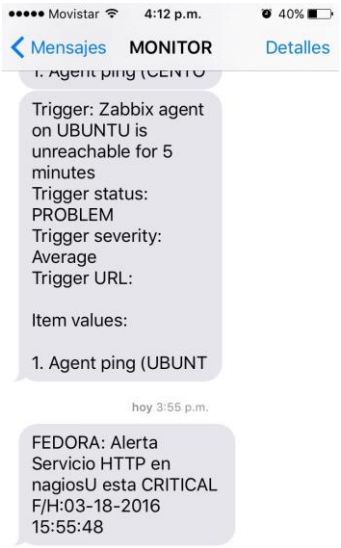


Fig. 4. Mensaje de Texto Recibido

5. CONCLUSIONES

El monitoreo de redes es una de las actividades más importantes hoy en día, las redes de datos están en constante crecimiento y cambio, surgen nuevas tecnologías y los administradores de red deben tener una línea de comunicación directa para conocer cuál es el estado de la red: estado normal, o uno que se considera un estado crítico o un estado de advertencia. Al analizar los resultados obtenidos en las pruebas realizadas y comparando las Tablas de Uso y Carga del CPU, y Memoria RAM, se concluye que la mejor distribución de las tres que se instalaron con NAGIOS es CENTOS. Es importante recalcar que los equipos de cómputo utilizados tenían algunas limitaciones en cuanto a memoria RAM, Disco Duro y Procesador, sin embargo el monitoreo se realizó correctamente. Con las características que existen en la actualidad esta actividad se lleva a cabo sin problemas.

6. AGRADECIMIENTOS

Agradecemos al Tecnológico Nacional de México, por el apoyo y financiamiento para llevar a cabo este proyecto bajo el registro **6253.17-P**.

7. REFERENCIAS

[1] López V. A., “Estudio de estándares de diseños físicos de LAN y su adecuación a la topología del lugar”, Revista Digital Universitaria, [en línea], disponible http://www.revista.unam.mx/vol.5/num5/art28/art28.htm, 5 de Enero de 2017.

[2] Galstad E., “Nagios Core”, NAGIOS [en línea], disponible: https://assets.nagios.com/downloads/nagioscore/docs/nagioscore/3/en/about.html#whatis, 24 de Abril de 2017.

[3] Itcentralstation, “Network Monitoring Software”, IT Central Station, 1771, Mayo 2017, 35.

[4] Altamirano C. A. 2005. “Monitoreo de Recursos de Red”, Seminario ADMIN-UNAM, Ciudad de México.

[5] InformaticaModerna.com, “Celulares Modernos y Samrthphone”, [en línea], disponible: http://www.informaticamoderna.com/Celular.htm, 5 de Enero de 2017.

[6] Cihar M. “Gammu and Wammu”, [en línea], disponible: http://es.wammu.eu/gammu/, 5 de Enero de 2017.

[7] Nagios Enterprises, “NAGIOS CORE”, [en línea], disponible: https://assets.nagios.com/downloads/nagioscore/docs/nagioscore/3/en/objectdefinitions.html, 5 de Enero de 2017.

[8] StackExchange, “Serverfault”, [en línea], disponible: https://serverfault.com/questions/422377/how-to-configure-nagios-realtime-sms-alert, 5 de Enero de 2017.