

DetECCIÓN incidencias y emisión alarmas en la gestión de inventarios de red

Julio Antonio Hernández Pérez

Universidad de las Ciencias Informáticas, La Habana, Cuba, jhperez@uci.cu

Yoanni Ordoñez Leyva

Universidad de las Ciencias Informáticas, La Habana, Cuba, yordones@uci.cu

Ernesto Avilés Vázquez

Universidad de las Ciencias Informáticas, La Habana, Cuba, aviles@uci.cu

RESUMEN

El Gestor de Recursos de Hardware y Software es una plataforma que permite realizar el proceso de captura, análisis y consulta de la información de los activos informáticos inventariados en red de computadoras que tengan instalados sistemas operativos GNU/Linux o Windows. Cuenta con el módulo de alarmas y acciones ante incidencias encargado de la detección de cambios no deseados en los inventarios (incidencias), realización de acciones, emisión de alarmas y confección reportes sobre las incidencias detectadas. Para el desarrollo del módulo se hizo uso de los estilos arquitectónicos: Modelo-Vistas-Plantillas y basado en componentes. Se utilizó para la implementación el lenguaje de programación *Python* aprovechando sus potencialidades para el desarrollo de aplicaciones web, la captura de imágenes y el envío de mensajes vía correo electrónico, chat y SMS. La plataforma mejora la gestión de incidencias en inventarios de red con respecto a varias de las principales herramientas dedicadas esta actividad, definiendo una vía más flexible para determinar qué cambios en las computadoras inventariadas son considerados incidencias.

Palabras claves: emisión de alarmas, gestión de incidencias, inventarios de red

ABSTRACT

The Resource Manager Hardware and Software platform allows to capture, to analyze and to retrieval hardware and software information in a network with GNU/Linux or Windows installed on the computers. The alarms and actions for incidents module detects changes in inventories (incidences), perform actions, issue alarms and generates reports about changes detected. This module took advantages from architectural styles Model Views Templates and Component-Based. It was development in Python programming language so it uses libraries for creating web applications, capturing images and sending messages via e-mail, chat and SMS. The platform improves inventory management network incidents regarding several major tools dedicated this activity, defining a more flexible way to determine what changes are considered incidents inventoried computers.

Keywords: incident management, issuing alarms, network inventory

1. INTRODUCCIÓN

Los inventarios tienen como objetivo la fiscalización, identificación y categorización de los recursos tangibles de las organizaciones. Una rama particular de los inventarios son los inventarios de red, estos son controles periódicos que se realizan al hardware y al software de los activos de una red de computadoras. Un sistema de gestión de inventario de hardware y software es una aplicación con soporte de datos, que acumula información sobre los activos informáticos en una red de ordenadores (Vidal, 2012). Existen varias aplicaciones dedicadas a la

realizaciones de estos controles periódicos dentro de ellas destacan: NetSupportDNA (NetSupport, 2012) y OCS-Inventory NG (MODx, 2011) como las principales en el mundo propietario y libre respectivamente.

Generalmente estas aplicaciones se caracterizan por implementar la arquitectura cliente-servidor y realizar inventarios con frecuencias predefinidas por el usuario o cuando son demandados al instante por este. Estos sistemas cuentan con una consola de administración donde se procesan y muestran, en forma de reportes, los inventarios de red. Además cuentan con una serie de colectores instalados en las computadoras inventariadas que se encargan de la lectura de la información.

No obstante los usuarios de este tipo de aplicaciones manifiestan inconformidades con el funcionamiento de los Sistemas de Gestión de Inventarios de Red (SGIR). Dentro de estas inconformidades aparecen la no detección de anomalías en los inventarios (incidencias), como puede ser el cambio o pérdida de algún dispositivo de hardware o la presencia de software no autorizado o con licencias caducas en las computadoras inventariadas.

Por otro lado, pocas aplicaciones de inventario de red envían alarmas cuando existen anomalías en los inventarios. Solo se utiliza la vía del correo electrónico, dejando de tomar en cuenta otras como: los mensajes SMS y la mensajería instantánea (xmpp). Regularmente este tipo de aplicaciones al detectar cambios en los inventarios no realizan acciones que mitiguen los efectos negativos de estos cambios en las computadoras.

Tomando en cuenta todo lo anterior en la Universidad de la Ciencias Informáticas (UCI) se desarrolló un sistema encargado de realizar inventarios de red conocido como Gestor de Recursos de Hardware y Software (GRHS). Dentro de los módulos de dicho sistema existe uno dedicado a la gestión de incidencias, realización de acciones y la emisión de alarmas ante anomalías conocido como Módulo de Alarmas y Acciones ante Incidencias (MAAI).

El objetivo de este artículo es describir el funcionamiento, arquitectura y las tecnologías utilizadas en la construcción del MAAI. A continuación en la sección **Conceptos Generales** que definirán algunas categorías de importancia en el contexto la gestión de incidencias en GRHS. Luego en el acápite **Gestores de Inventarios de Red** se explican funcionalidades de algunos SGIRs haciendo una breve comparación entre estos para seguidamente explicar **las funcionalidades y tecnologías utilizadas en el MAAI**. En la sección **Arquitectura** se expone la arquitectura del MAAI en el sistema GRHS y en la sección **Procesos de detección de incidencias en los inventarios de red** se reflejará brevemente como se llevan a cabo las actividades de este proceso en GRHS a través de MAAI. Por último se presentarán algunos **Resultados** de las pruebas funcionales realizadas al módulo y algunas consideraciones como **Conclusiones** del desarrollo realizado.

2. CONCEPTOS GENERALES

Con el objetivo de propiciar un mejor entendimiento de la solución tecnológica expuesta en este artículo se hace necesario la definición de algunos conceptos que son manejados en el mismo. Ellos son:

Inventario de red: controles periódicos que se realizan al hardware y al software de los activos de una red de computadoras.

Incidencia: entidad principal dentro del funcionamiento de MAAI. La incidencia es la forma de describir qué cambios dentro de los inventarios son considerados como anomalías.

Acción: actividad que se ejecuta de forma automática en las computadoras inventariadas cuando es detectada una incidencia.

Imagen: tipo de acción que consiste en capturar de forma automática imágenes en computadoras objeto de incidencia.

Alarma: mensaje de alerta emitido por GRHS luego de detectada la ocurrencia de una incidencia.

Control: variante de acción que no está asociado a incidencia alguna y se ejecuta en el momento que el administrador de GRHS lo requiera, ejemplos: apagar, reiniciar, suspender e hibernar la computadora.

3. GESTORES DE INVENTARIOS DE RED

3.1 NETSUPPORT DNA

Es la herramienta líder dentro de los SGIR en el mundo propietario. Permite la realización de inventarios del hardware y el software y la generación de informes de los inventarios realizados. Es compatible con la mayoría de los sistemas operativos de la familia de Microsoft además de Suse, Fedora y Redhat dentro de la familia GNU/Linux. Implementa la arquitectura cliente-servidor. Cuenta con funcionalidades para notificación de alarmas vía correo electrónico. Permite la gestión y actualización de licencias de software y es capaz de detectar cambios en el hardware, software, energía y consumo de CPU de las computadoras inventariadas. Esta herramienta no cuenta con funcionalidades para la configuración de otros tipos de incidencias pues los cambios a detectar no son modificables (NetSupport, 2012).

3.2 VEO

Es un SGIR con varias funcionalidades dentro de las que destacan: el inventario de hardware y software, la administración remota de computadoras, toma el control de los periféricos mouse y teclado, toma imágenes de las estaciones de trabajo y la desinstalación de programas no autorizados. Por otro lado esta herramienta no es multiplataforma, sus versiones solo son compatibles con los sistemas operativos de Microsoft. No es capaz de detectar cambios en los inventarios de red e implementa la arquitectura cliente-servidor (QMA, 2012).

3.3 OCS-INVENTORY NG

Es un SGIR de software libre. Permite la realización de inventarios de hardware y software e implementa la arquitectura cliente-servidor. Esta aplicación es multiplataforma y posee funcionalidades para la realización de reportes sobre los inventarios siendo capaz de enviarlos a través del correo electrónico. Utiliza varias tecnologías como: PHP, Perl y MySQL. Esta herramienta no tiene implementaciones para la detección cambios en los inventarios, ni para la emisión de alarmas dada la ocurrencia de estos cambios (MODx, 2011).

3.4 CACIC (CONFIGURADOR AUTOMÁTICO Y COLECTOR DE INFORMACIONES COMPUTACIONALES)

Es un SGIR a código abierto y multiplataforma. Utilizan varias tecnologías como: PHP, Python, Delphi™ y el gestor de base de datos MySQL. Es capaz de generar alarmas cuando existe un cambio en la ubicación física de las computadoras inventariadas y detecta cambios en los inventarios de los componentes de hardware. CACIC no posee funcionalidades para la configuración de nuevas alarmas o la detección de otros cambios en el software de las computadoras (Dataprev, 2011).

3.5 GRHS (GESTOR DE RECURSOS DE HARDWARE Y SOFTWARE)

Es un SGIR que implementa la arquitectura cliente-servidor. Realiza inventarios de hardware y software, monitorización de instalación de programas y conexión de dispositivos USB. GRHS cuenta con un conjunto de colectores capaces de detectar cambios en la configuración de los equipos, así como de ejecutar acciones y alarmas una vez identificados estos cambios. Los cambios a detectar, acciones y alarmas son configurables. Esta plataforma con tres aplicaciones: (1) *gclient* que se encarga de la gestión, detección cambios y notificaciones de los inventarios de las computadoras, (2) *gserver* responsable de la recepción y el almacenamiento de los inventarios, emisión de órdenes y envío de configuraciones a los colectores (*gclient*) y (3) *gadmin* aplicación web donde se puede consultar la información de los inventarios y realizar las configuraciones del sistema. GRHS está disponible para los sistemas operativos Microsoft y las distribuciones de Debian y Ubuntu de GNU/Linux a partir de las versiones 6 y 12.04 respectivamente.

3.6 COMPARACIÓN ENTRE LOS SGIR

A continuación se muestra una tabla donde son comparados los SGIR antes mencionados en cuanto a: si son multiplataforma, si detectan cambios en el hardware o software de las computadoras, si emiten alarmas cuando son identificados estos cambios, si los cambios a detectar y alarmas a emitir son configurables y por último el total de características que cumple cada SGIR.

Tabla 1: Comparación entre SGIR

SGIR	Multiplataforma	Cambios en Hardware	Cambios en Software	Emisión de alarmas	Configurable	Total
Netsupport DNA	x	x	x	x	-	4/5
VEO	-	-	-	-	-	0/5
OCS-Inventory NG	x	-	-	-	-	1/5
CACIC	x	x	-	x	-	3/5
GRHS	x	x	x	x	x	5/5

De forma general los SGIR de la tabla son aplicaciones multiplataforma que detectan la ocurrencia cambios en el hardware de las computadoras inventariadas. En menor medida detectan cambios en el software y emiten alarmas. Como aspecto significativo se tiene que únicamente GRHS permite configurar qué cambios dentro del inventario son considerados incidencias. Este último aspecto es importante según nuestro criterio pues permite a los usuarios de estas herramientas personalizar la gestión de incidencias y la configuración de alarmas dentro de sus redes siendo este un aspecto importante en el control de inventarios de red. La gestión de incidencias en los inventarios de red en GRHS se realiza a través del MAAI, eje fundamental de este artículo. A continuación se describen sus funcionalidades.

4. FUNCIONALIDADES Y TECNOLOGÍAS

4.1 FUNCIONALIDADES DE MAAI

El MAAI posee las siguientes funcionalidades:

- Detección de cambios en los inventarios.
- Realización de acciones ante incidencias.
- Emisión de alarmas ante incidencias.
- Realización de reportes sobre las incidencias detectadas, acciones realizadas y alarmas emitidas.

A continuación se muestra la Tabla 2: *Resumen de acciones, alarmas y estadísticas del MAAI* que se resume las acciones, vias de alarmas y estadísticas del MAAI.

Tabla 2: Resumen de acciones, alarmas y estadísticas del MAAI

Acciones	Alarmas	Reportes
<ul style="list-style-type: none"> • Captura de imagen de escritorio y con <i>webcam</i>. • Apagar, reiniciar, suspender e hibernar computadoras (controles). • Bloqueo de cuentas de usuarios. 	<ul style="list-style-type: none"> • Correo electrónico. • Chat (<i>xmpp</i>). • SMS. 	<ul style="list-style-type: none"> • Total de incidencias detectadas. • Incidencias de hardware. • Incidencias de software. • Acciones realizadas. • Controles realizados. • Imágenes capturadas.

4.2 TECNOLOGÍAS UTILIZADAS EN EL DESARROLLO DEL MAAI

Para la implementación del MAAI fue utilizado como lenguaje de programación Python en su versión 2.7 (Gonzalez Duque, 2006). Dentro de las bibliotecas de Python empleadas para la captura de imágenes en sistemas

operativos GNU/Linux se encuentran: *python-gst* (Persson, Jens, 2010) y *python-gtk* (The GNOME Project and PyGTK Team, 2010) y en los sistemas operativos Windows *python-win32* (Python Software Foundation, 2011) y *vidcap* (Gritsch, 2013). El bloqueo de cuentas de usuarios, apagado, reiniciado y demás acciones en ambos sistemas operativos fue desarrollado utilizando implementaciones de la biblioteca estándar del lenguaje para la ejecución de comandos.

En el caso de la implementación del envío de alarmas se utilizaron las bibliotecas: *python-smtp* (Python Software Foundation, 2013) para el envío de alarmas por correo electrónico, *python-xmpp2* (Python Software Foundation, 2011) para el envío de mensajes vía chat y *python-gammu* (Čihař, 2010) para el envío de alarmas vía SMS. Para mostrar los reportes antes mencionados se utilizó el *framework* web *Python Django 1.4* (Django Software Foundation, 2005).

5. ARQUITECTURA

GRHS es una plataforma con arquitectura N-tiers que cuenta con tres aplicaciones: *gclient*, *gserver* y *gadmin*. En las tres aplicaciones existen componentes, que de forma coordinada, se dedican a la gestión de incidencias en los inventarios y la emisión de alarmas y ejecución de acciones cuando son detectadas las incidencias. En la Figura 1: Arquitectura de GRHS y el MAAI se muestra la ubicación de los componentes del MAAI en las aplicaciones de la plataforma.

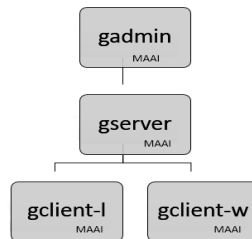


Figura 1: Arquitectura de GRHS y el MAAI

En la aplicación *gadmin* el componente del módulo MAAI está estructurado teniendo como base la arquitectura del *framework* web *Python Django 1.4* (Django Software Foundation, 2005). Este *framework* sigue el estilo arquitectónico Modelo-Vista-Plantilla. El componente MAAI de *gadmin* está compuesto por tres aplicaciones *django*: *action*, *configuration* e *incidence* como se muestra en la figura 2. Cada una de estas aplicaciones está compuesta por tres capas, una dedicada a la presentación de la información (*template*), otra dedicada a las vistas (*view*) y una última que agrupa los modelos (*model*) que constituyen la representación de los datos en el sistema.

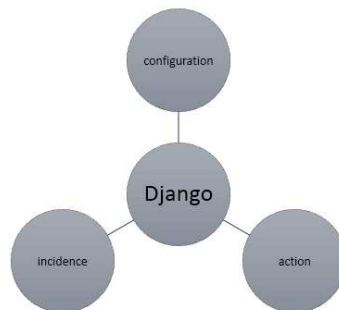


Figura 2: Arquitectura del componente MAAI en gadmin

En la aplicación *gserver* el componente MAAI tiene cinco aplicaciones *django*, estas son: *actions*, *alarms*, *controls*, *incidence* y *pictures* como se muestra en la Figura 3: Arquitectura del componente MAAI de *gserver*. Cada una de estas aplicaciones posee dos capas: *view* que utilizando la aplicación *django-restframework* (DabApps, 2012) muestra los datos que son representados por los modelos de la capa *model*.

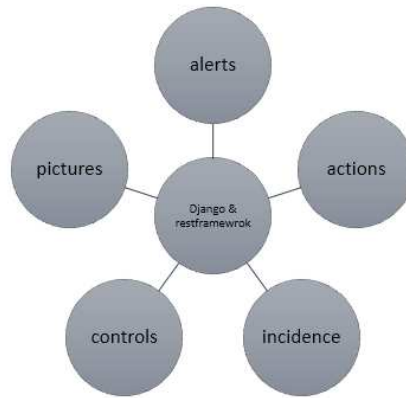


Figura 3: Arquitectura del componente MAAI de gserver

La aplicación *gclient* tiene una arquitectura basada en componentes de ahí que posee varios *plugins* que en su conjunto conforman MAAI dentro de *gclient*, estos son: *actions*, *control*, *incidence* y *pictures* y su funcionamiento está coordinado por *pydi*. El *pydi* es un componente de software desarrollado por el equipo de desarrollo de GRHS encargado de las funciones generales de la aplicación *gclient*, tales como: tratamiento de trazas, comunicación con *gserver* y manejo de excepciones. En la Figura 4: Arquitectura del MAAI en *gclient* se muestran la relación de *pydi* con los *plugins* antes mencionados de *gclient*.

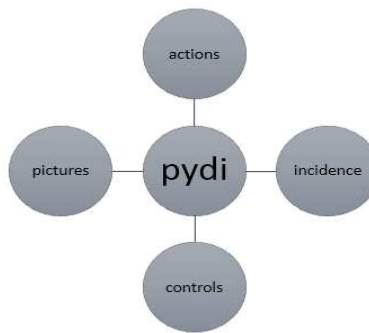


Figura 4: Arquitectura del MAAI en gclient

6. PROCESOS DE DETECCIÓN DE INCIDENCIAS EN LOS INVENTARIOS DE RED.

El proceso de gestión de incidencias automatizado en el MAAI dentro de la plataforma GRHS están constituidos de forma general por las tareas: (1) definir y categorizar incidencia, (2) definir alarmas y acciones a ejecutar ante la incidencia detectada, (3) detectar incidencia, (4) ejecutar acciones y (5) emitir alarmas tal y como se muestra en la figura siguiente.

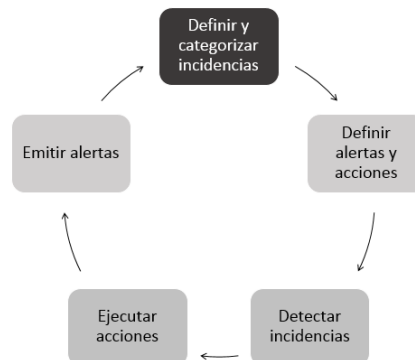


Figura 5: Tareas del MAAI

Como primer paso y a través de la consola de administración de GRHS los usuarios de la herramienta configuran que cambios dentro de los inventarios serán considerados como incidencias teniendo en cuenta los siguientes aspectos: componente elemento de inventario (de hardware o software) objeto del cambio, estado refiere a cuál fue el cambio en concreto, sustracción o adicción del componente, nivel es una etiqueta creada por el usuario que permite clasificar a un conjunto de incidencias, tipo está en dependencia del componente y puede ser de hardware y software.

Luego de definida y categorizada la incidencia se pasa entonces a definir qué acciones y alarmas se llevarán a cabo cuando esta sea detectada. Según el nivel de la incidencia se ejecutan determinadas acciones y alarmas. De ahí que a varias incidencias se le puedan asociar las mismas acciones y alarmas o no. En la tabla siguiente se detalla las acciones y alarmas a ejecutar cuando se detecten hipotéticamente incidencias categorizadas como emergencia. Estas configuraciones también se realizan en la consola de administración de GRHS.

Tabla 3: Definición de alarmas y acciones para incidencias emergencia

Nivel	Acciones	Alarmas
<ul style="list-style-type: none"> Emergencia 	<ul style="list-style-type: none"> Imagen de escritorio Imagen con <i>webcam</i> 	<ul style="list-style-type: none"> Correo electrónico Chat SMS

Después de realizar las configuraciones del sistema estas son enviadas a los colectores instalados en las computadoras inventariadas (*gclient*). Al ejecutar un nuevo inventario los colectores comparan el resultado del último inventario con el almacenado en el servidor (*gserver*); si se detecta la ausencia o cambio del mouse entonces el colector captura imágenes del escritorio y con la *webcam* de la computadora. Estas imágenes se envían a *gserver* que registra la incidencia, el nuevo inventario y emite las alarmas configuradas para la categoría de la incidencia detectada. Para el ejemplo de la Tabla 3 las alarmas que serán emitidas detallan las características de la anomalía detectada. La Tabla 5 ejemplifica la información que contienen los mensajes que se envían. En el caso del dato *Host* es un identificador que el GRHS asigna a las computadoras durante la instalación para identificar las mismas y la fecha es el instante de tiempo en el cual fue detectada la incidencia.

Tabla 4: Datos del mensaje de alerta para una incidencia detectada.

Datos	Valor
Host	MB58880
Componente	Mouse
Estado	Eliminado
Nivel	Emergencia
Fecha y Hora	06-04-12 22:30

Luego de realizadas todas estas tareas son actualizados los reportes de incidencias, alarmas y acciones en la aplicación web. De este modo se ha descrito el funcionamiento básico del MAAI en la plataforma GRHS.

7. RESULTADOS DE LAS PRUEBAS DE VERIFICACIÓN

Para probar el funcionamiento se la gestión de incidencias en GRHS se realizó un despliegue de la plataforma en las instalaciones de la Facultad 2 de la UCI. El mismo cuenta con más de 70 computadoras alojadas en 6 locales y con 7 subredes. Estas computadoras tienen gran diversidad en cuanto a las características de los dispositivos de hardware que poseen y al software que tienen instalado. El colector *gclient* se instaló en 30 computadoras. GRHS fue configurado con 3 tipos de incidencias: (1) alta para la cual se configuró la captura de imágenes del escritorio y la emisión de mensajes vía correo electrónico como alarmas, (2) media para la cual se configuró el bloqueo de cuentas de usuarios y (3) baja para lo cual se configuró la emisión de una alarma vía chat. Para ello se definió como incidencia: la conexión de dispositivo de almacenamiento, la instalación de programas y la desconexión de dispositivo de almacenamiento. Los resultados se muestran en la siguiente tabla.

Tabla 5 Resultados la gestión de incidencias

Variables	Valores
-----------	---------

Total de incidencias	2742
Total de incidencias hardware	2729
Total de incidencias software	13
Total de acciones	6
Total de alarmas	2703
Total de imágenes	2

Los resultados obtenidos fueron observados en una semana de despliegue. Demuestran la efectividad de realizar todas las tareas de gestión de incidencias a través de la aplicación. Este aspecto permite mantener el control del estado de los activos informáticos de una red de computadoras y de los cambios no deseados que pueden ocurrir con la configuración de los equipos. Empleados y directivos de la institución tienen una crítica positiva del funcionamiento de la herramienta en particular de la gestión de incidencias que se realiza a través del MAAI.

8. CONCLUSIONES

Dentro de GRHS, MAAI flexibiliza la definición de sucesos anómalos en los inventarios, mejorando así la gestión de incidencias con respecto al resto de los SGIR. Por otro lado permite la configuración de mensajes de alarmas cuando son detectados las incidencias, utilizando nuevas vías no explotadas por el resto de los SGIR como los SMS y los mensajes vía chat. La arquitectura basada en componente de la aplicación *gclient* permite añadir o eliminar de forma sencilla nuevas acciones y vías de alarmas a ejecutar ante las incidencias. El uso de GRHS representa un ahorro de tiempo y recursos humanos pues una vez configurados las incidencias, acciones a ejecutar y alarmas a emitir, este actúa de forma autónoma. El sistema es de código abierto por lo que puede ser mejorado por otros desarrolladores y adaptado a las necesidades de otras organizaciones.

REFERENCIAS

- Čihař, M. 2010. Python gammu API. *Python gammu API*. [En línea] 2010. [Citado el: 3 de febrero de 2013.] <http://wammu.eu/docs/manual/python>.
- Django Software Foundation. 2005. Django. *Django*. [En línea] 2005. [Citado el: 21 de febrero de 2013.] <https://www.djangoproject.com/>.
- DabApps. 2012. Django REST Framework. *Django REST Framework*. [En línea] 2012. [Citado el: 22 de febrero de 2013.] <http://django-rest-framework.org/>.
- Dataprev. 2011. Configurador Automático e Coletor de Informações Computacionais. *Configurador Automático e Coletor de Informações Computacionais*. [En línea] 2011. [Citado el: 20 de enero de 2013.] http://www.softwarepublico.gov.br/spb/ver-comunidade?community_id=3585.
- Gonzalez Duque, R. 2006. *Python para Todos*. Madrid, España : Creative Commons Reconocimien-to 2.5, 2006. págs. 6-9.
- Gritsch, M. 2013. A Win32 Python Extension for Accessing Video Devices. *A Win32 Python Extension for Accessing Video Devices*. [En línea] 10 de enero de 2013. <http://videocapture.sourceforge.net>.
- MODx. 2011. OCS Inventory NG. *OCS Inventory NG*. [En línea] 2011. [Citado el: 16 de enero de 2013.] <http://www.ocsinventory-ng.org/en/>.
- NetSupport. 2012. NetSupport DNA. *NetSupport DNA*. [En línea] 2012. [Citado el: 23 de enero de 2013.] <http://www.netsupportdna.com>.
- Persson, Jens. 2010. Python GStreamer Tutorial. *Python GStreamer Tutorial*. [En línea] 2010. [Citado el: 9 de enero de 2013.] <http://pygstdocs.berlios.de/pygst-tutorial/index.html>.
- Python Software Foundation. 2013. smtplib — SMTP protocol client. *smtplib — SMTP protocol client*. [En línea] 6 de febrero de 2013. <http://docs.python.org/library/smtplib.html>.
- Python Software Fundation. 2011. Python for windows. *Python for windows*. [En línea] 2011. [Citado el: 11 de febrero de 2013.] <http://www.python.org/getit/windows/>.
- Python Software Fundation. 2011. xmpp2 0.4. *xmpp2 0.4*. [En línea] 2011. [Citado el: 21 de enero de 2013.] <http://pypi.python.org/pypi/xmpp2>.

- QMA. 2012. Inventario de Hardware y Software con el monitoreo de estaciones de red -VEO Ultimate. *Inventario de Hardware y Software con el monitoreo de estaciones de red -VEO Ultimate*. [En línea] 2012. [Citado el: 15 de enero de 2013.] <http://www.veo.com.mx>.
- The GNOME Project and PyGTK Team. 2010. PyGTK. *PyGTK*. [En línea] 2010. [Citado el: 2 de febrero de 2013.] <http://www.pygtk.org>.
- Vidal, J. 2012. *Fundamentos de gestión y control de inventarios*. Holguín, Cuba : s.n., 2012.