

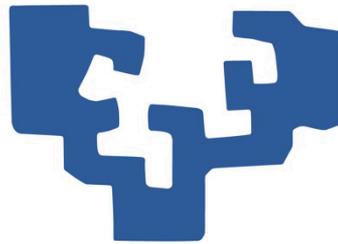
Análisis de Plataforma de Monitorización de Redes y Sistemas y Diseño de Mejoras

Proyecto Fin de Grado (PFG) de Grado en Ingeniería Informática, especialidad en Ingeniería de Computadores

Centro: Facultad de Informática de San Sebastián (FISS)

Empresa: COM&MEDIA Proyectos y Servicios, S.L.

eman ta zabal zazu



Universidad
del País Vasco

Euskal Herriko
Unibertsitatea

Alumno: Ander Cid Rivas

Director (FISS): José Miguel Alonso

Codirector (COM&MEDIA): Mikel Aguirre Zulaika

Junio de 2013

Esta publicación se encuentra bajo **Licencia Creative Commons BY-NC-ND 3.0** según se detalla en <http://creativecommons.org/licenses/by-nc-nd/3.0>. Su autoría corresponde a Ander Cid Rivas, de DNI 72540114-T.

Usted es libre de:

Compartir -- copiar, distribuir, ejecutar y comunicar públicamente la obra.

Bajo las siguientes condiciones:

Atribución -- Debe reconocer los créditos de la obra de la manera especificada por el autor o el licenciante (pero no de una manera que sugiera que tiene su apoyo o que apoyan el uso que hace de su obra).

No Comercial -- No puede utilizar esta obra para fines comerciales.

Sin Obras Derivadas -- No se puede alterar, transformar o generar una obra derivada a partir de esta obra.

Entendiendo que:

Renuncia -- Alguna de estas condiciones puede no aplicarse si se obtiene el permiso del titular de los derechos de autor.

Dominio Público -- Cuando la obra o alguno de sus elementos se halle en el dominio público según la ley vigente aplicable, esta situación no quedará afectada por la licencia.

Otros derechos -- Los derechos siguientes no quedan afectados por la licencia de ninguna manera:

- Los derechos derivados de usos legítimos u otras limitaciones reconocidas por ley no se ven afectados por lo anterior.
- Los derechos morales del autor;
- Derechos que pueden ostentar otras personas sobre la propia obra o su uso, como por ejemplo derechos de imagen o de privacidad.

Puede consultar las bases de la licencia completa en la siguiente URL:

<http://creativecommons.org/licenses/by-nc-nd/3.0/legalcode>

CONVENIO APLICABLE SOBRE PROPIEDAD INTELECTUAL

La propiedad de los derechos intelectuales, productos (si los hubiera) y los medios de difusión del trabajo serán los siguientes:

- El alumno ostenta el derecho de presentar, en eventos propios de su actividad en los que participe (cursos, seminarios, congresos, etc.), los resultados obtenidos en el proyecto final de carrera. Del mismo modo podrá hacer publicidad de dichos resultados en las publicaciones que considere oportunas. En cualquier caso siempre deberá referenciar a la empresa en la que ha realizado el proyecto.

- En ningún caso la publicación o divulgación de los resultados del proyecto conllevará un grado de detalle tal que implique o pueda implicar su desarrollo material por parte de terceros.

- Por otra parte, en cualquier trabajo, publicación o producto que se obtenga en el futuro a partir de los resultados del proyecto fin de carrera por parte de la empresa se referenciará de forma expresa al alumno como copartícipe en la obtención de dichos resultados.

- Ni la empresa, ni el alumno ni la universidad podrán comercializar el producto sin mutuo acuerdo.

- Una vez finalizado el proyecto, se reunirán las partes Implicadas con el fin de llegar a un acuerdo sobre el software creado, y siempre en base a los siguientes puntos:

Primero: *La empresa en cualquier caso se quedará con una copia del software, y se reservará el uso interno del mismo.*

Segundo: *La empresa podrá adquirir todos los derechos del software, llegando previamente a un acuerdo económico.*

Tercero: *En caso de no interesar a la empresa la adquisición de los derechos, la autora podrá utilizar el software según su conveniencia, quedándose la empresa con una copia para su uso interno, según se Indica en el punto primero de las bases.*

ÍNDICE GENERAL

1.-PREÁMBULO	15
1.1.- Objetivos	15
1.2.- Experiencia Previa y Motivación	16
Referencias/Enlaces del Capítulo	17
2.- MARCO DE TRABAJO INICIAL	19
3.- PRIMERA FASE	21
3.1.- Análisis de Pandora FMS	21
3.1.1.- Licencias del Uso	21
3.1.2.- Arquitectura. Elementos de una Instalación Básica.	21
3.1.3.- Funcionamiento de un Chequeo	24
3.1.3.1.- Modelo de Capas	24
3.1.4.- Desarrollo de <i>Plug-Ins</i>	26
3.1.4.1.- Dentro de un Agente Software	26
3.1.4.2.- Dentro del Servidor de <i>Plug-Ins</i>	27
3.1.5.- Generación de Alertas	28
3.1.5.1.- Protección en Cascada	30
3.1.6.- <i>Reporting</i>	30
3.1.6.1.- Generación de Gráficos	30
3.1.6.2.- Generación de Informes	31
3.1.7.- Mapas de Red	32
3.1.8.- Gestión de Usuarios y Permisos	33
3.2.- Instalación de Pandora FMS en un Entorno Virtual	33
3.2.1.- VM de Pandora FMS Server	34
3.2.1.1.- Instalación de Dependencias	34
3.2.1.2.- Instalación de Pandora FMS Server	35
3.2.1.3.- Configuración de Email	36
3.2.1.4.- Configuración de Servidor de Recepción de Traps SNMP	36
3.2.2.- VM de Servidor Linux	37
3.2.2.1.- Instalación y Configuración de Agente	37
3.2.3.- VM de Servidor Windows	37
3.2.3.1.- Instalación y Configuración de Agente	38
Referencias/Enlaces del Capítulo	38
4.- SEGUNDA FASE	39
4.1.- Equipamiento WiMax Alvarion	41
4.1.1.- Análisis de Dispositivos	41
4.1.1.1.- Alvarion BreezeNET B	41
4.1.1.2.- Alvarion BreezeULTRA	42
4.1.2.- Dispositivos de Prueba	44

4.1.3.- Creación de Agentes	44
4.1.4.- Desarrollo de Módulos	44
4.1.4.1- Módulos BreezeNET	45
4.1.4.2.- Módulos BreezeULTRA	47
4.1.5.- Configuración de Alertas	49
4.1.6.- Evaluación de Soluciones	49
4.1.6.1.- Contraste de Información con el Servidor Web BreezeULTRA	49
4.1.6.2.- Cierre del Túnel VPN	50
4.2.- Equipamiento Networking Industrial Hirschmann	53
4.2.1.- Análisis de Dispositivos	53
4.2.1.1.- MACH102-8TP-R	53
4.2.1.2.- MACH1000 Full GE L3P PoE	53
4.2.2.- Dispositivos de Prueba	54
4.2.3.- Creación de Agentes	54
4.2.4.- Desarrollo de Módulos	55
4.2.4.1.- Módulos MACH1000	55
4.2.4.2.- Módulos MACH102	59
4.2.5.- Configuración de Alertas	61
4.2.6.- Evaluación de Soluciones	61
4.2.6.1.- Limpieza de Ficheros No Utilizados	61
4.2.6.2.- Pérdida de Paquetes (Plug-In)	61
4.2.6.3.- Uso de Potencia PoE (Plug-In)	63
4.2.6.4.- Contraste de Información con el Servidor Web	64
4.2.6.5.- Fuentes de Alimentación	64
4.3.- Servidores HP ProLiant	65
4.3.1.- Análisis de Dispositivos	65
4.3.1.1.- HP ProLiant DL360	65
4.3.1.2.- HP ProLiant DL380	66
4.3.1.3.- Plataforma HP Integrated Lights-Out (iLO)	66
4.3.1.4.- Estándares WBEM y CIM	67
4.3.2.- Dispositivos de Prueba	68
4.3.2.1.- Instalación y Puesta en Marcha de Servidor HP ProLiant DL360 G5	68
4.3.2.2.- Configuración de WBEM/CIM	70
4.3.3.- Creación de Agentes	70
4.3.4.- Desarrollo de Módulos	70
4.3.4.1.- Vía IPMI	70
4.3.4.2.- Vía WBEM/CIM	72
4.3.5.- Configuración de Alertas	73
4.3.6.- Evaluación de Soluciones	74
4.3.6.1.- Contraste de Información con la Plataforma iLO3	74
4.3.6.2.- Apagado del Chasis	74
4.3.6.3.- Desconexión de una Fuente de Alimentación	74
4.3.6.4.- Desconexión de un Módulo de Memoria RAM	75
4.3.6.5.- Conexión de una Tarjeta de Red	75
4.3.6.6.- Desconexión de un Disco Duro	75
4.4.- Sistemas de Almacenamiento EMC²	77
4.4.1.- Análisis de Dispositivos	77
4.4.1.1.- VNX Series	77
4.4.1.2.- VNXe Series	78
4.4.1.3.- Unisphere Manager	79
4.4.2.- Dispositivos de Prueba	79
4.4.3.- Creación de Agentes	79
4.4.4.- Desarrollo de Módulos	80
4.4.5.- Configuración de Alertas	82

4.4.6.- Evaluación de Soluciones	82
4.4.6.1.- Contraste con EMC Unisphere Web Console	82
4.5.- Entorno de Virtualización VMware	85
4.5.1.- Análisis del Entorno	85
4.5.1.1.- Tecnología vMotion y High Availability (HA)	85
4.5.1.2.- VMware ESX	86
4.5.1.3.- VMware ESXi	86
4.5.2.- Entornos de Prueba	87
4.5.2.1.- Activación de snmpd sobre ESXi 5.1	87
4.5.2.2.- Instalación de Máquinas Virtuales	88
4.5.3.- Creación de Agentes	88
4.5.4.- Desarrollo de Módulos	88
4.5.5.- Configuración de Alertas	90
4.5.6.- Evaluación de Soluciones	90
4.5.6.1.- Apagado y Encendido de VMs	90
4.5.6.2.- Exceso de Reserva de Memoria RAM	91
4.5.6.3.- Contraste con vSphere Client	91
4.5.6.4.- CPU Stress Test	91
4.5.6.5.- Disk Stress Test	92
4.6.- Oracle Database Server	93
4.6.1.- Análisis del Sistema	93
4.6.1.1.- Arquitectura Básica	93
4.6.1.2.- Acceso a Información de Estado y Rendimiento	94
4.6.2.- Entorno de Prueba	95
4.6.2.1.- Instancia Oracle	95
4.6.2.2.- Instalación de Cliente Oracle en Pandora Server	95
4.6.3.- Creación de Agentes	95
4.6.4.- Desarrollo de Módulos	96
4.6.4.1.- Módulos SQL*Plus	96
4.6.4.2.- Módulos Adaptados de Nagios Core	96
4.6.4.3.- Módulos Oracle NET	98
4.6.5.- Configuración de Alertas	98
4.6.6.- Evaluación de Soluciones	98
4.6.6.1.- Oracle Enterprise Manager Console	98
4.7.- Servidores de Aplicaciones Web Tomcat	101
4.7.1.- Análisis del Dispositivo	101
4.7.1.1.- Ficheros de Configuración	101
4.7.1.2.- Extracción de Información	102
4.7.2.- Entorno de Prueba	102
4.7.3.- Creación de Agentes	102
4.7.4.- Desarrollo de Módulos	103
4.7.4.1.- Módulos con Fichero Caché	103
4.7.4.2.- Módulos sin Fichero Caché	104
4.7.5.- Configuración de Alertas	104
4.7.6.- Evaluación de Soluciones	104
4.7.6.1.- Peticiones Masivas de Descarga de Páginas Estáticas	104
4.7.6.2.- Peticiones Masivas de Ejecución de Aplicaciones	105
4.7.6.3.- Parada de un Servicio Web	107
Referencias/Enlaces del Capítulo	109
5.- TERCERA FASE	117

5.1.- Marco de Uso	117
5.2.- Capacidad Nativa de Reporting	117
5.2.1.- Estructura de <i>Reporting</i>	117
5.2.2.- Análisis de Cumplimiento de <i>SLAs</i>	118
5.2.3.- Formatos de Salida	119
5.2.3.1.- Exportación de Reports	119
5.2.3.2.- Exportación de Información de Módulos	119
5.3.- API Externa de Pandora FMS	120
5.3.1.- Consideraciones de Seguridad	120
5.3.2.- Parámetros <i>GET</i> Adicionales	121
5.3.3.- Llamadas de Prueba	121
5.3.3.1.- Eventos del Sistema	121
5.3.3.2.- Todos los Agentes	122
5.3.3.3.- Módulos de un Agente	123
5.3.3.4.- Información de un Módulo	124
5.3.3.5.- Gráficos	124
Referencias/Enlaces del Capítulo	125
6.- CONCLUSIONES	127
ANEXO 1.- PLAN DE GESTIÓN	131
A1.1.- Equipo de Proyecto	131
A1.2.- Alcance del Proyecto	131
A1.2.1.- Producto Objetivo	131
A1.2.2.- Grupo de Interés	131
A1.2.3.- Restricciones Generales	132
A1.2.4.- Costes Financieros	132
A1.3.- Plan de Comunicaciones	133
A1.3.1.- Comunicaciones Externas	133
A1.3.2.- Difusión del PFG	133
A1.4.- Estructura de Desglose de Trabajo (EDT)	133
A1.5.- Planificación Temporal	136
A1.5.1.- Diagrama de <i>Gantt</i> e Hitos	136
A1.5.2.- Distribución de Carga por Semanas	137
A1.6.- Plan de Riesgos	138
A1.6.1.- Plan de Copias de Seguridad	139
ANEXO 2.- INFORME DE SEGUIMIENTO Y CONTROL	140
A2.1.- Incidencias	140
A2.2.- Alcance Final	141
A2.2.1.- Producto Final	141
A2.2.2.- Costes Financieros	142
A2.3.- Dedicación y Costes	142

A2.4.- Adquisiciones y Licencias	145
ANEXO 3.- SCRIPTS GENERADOS	147
A3.1.- Scripts Networking Industrial Hirschmann	147
perdidos.pl (original)	147
perdidos_s.pl	150
hirschmann_uso_poe.sh	153
A3.2.- Scripts Servidores HP ProLiant	154
hp_ipmi_chassis_power.sh	154
hp_ipmi_power_meter.sh	154
hp_ipmi_power_supply.sh	155
hp_ipmi_temp_fuentes.sh	155
hp_ipmi_temp_ram.sh	155
hp_ipmi_temp_ambiental.sh	156
hp_ipmi_temp_disk.sh	156
hp_ipmi_fan.sh	157
check_cim_hpq.py (modificado)	157
check_esxcli_hparray (modificado)	158
hp_wbem_chasis_estado.sh	160
hp_wbem_nic_estado.sh	161
hp_wbem_ps_estado.sh	161
hp_wbem_ram_cap.sh	162
hp_wbem_ram_estado.sh	162
hp_wbem_cpu_estado.sh	163
hp_wbem_ram_slots.sh	163
A3.3.- Scripts EMC² VNX/VNXe	164
emc_uemcli_ssd_estado.sh	164
emc_uemcli_alertas.sh	164
emc_uemcli_bat.sh	165
emc_uemcli_discos_estado.sh	166
emc_uemcli_mm_estado.sh	166
emc_uemcli_nas_info.sh	167
emc_uemcli_nas_uso.sh	167
emc_uemcli_pool_info.sh	168
emc_uemcli_pool_uso.sh	168
emc_uemcli_pools_estado.sh	169
emc_uemcli_port_estado.sh	169
emc_uemcli_ps_estado.sh	170
emc_uemcli_rep_info.sh	170
emc_uemcli_snaps_info.sh	171
emc_uemcli_sp_estado.sh	171
A3.4.- Scripts VMware ESXi	172
esxi_cpu_uso.sh	172
esxi_mem_uso.sh	172
esxi_ram_allocation.sh	173
esxi_disk_iostats.sh	174
A3.5.- Scripts Oracle Database Server	175
oracle_sqlplus.sh	175
oracle_db_estado.sh	177
oracle_listener.sh	178

A3.6.- <i>Scripts Tomcat Server</i>	179
tomcat_info.sh	179
tomcat_app.sh	182

1.-PREÁMBULO

1.1.- Objetivos

Con este proyecto se pretende profundizar en el campo de la monitorización de equipos y servicios en red, y más precisamente, en la explotación (y posiblemente, desarrollo) de funcionalidades de una herramienta de monitorización concreta: *Pandora FMS Open Source*.



En el desarrollo del proyecto se presentan distintas necesidades de monitorización y *reporting* de la empresa para las que se debe hallar una solución, ya sea

- explotando las funcionalidades de *Pandora FMS*,
- integrando módulos de terceros (con cumplimiento de las licencias o limitaciones aplicables),
- adaptando el código de *Pandora FMS* o de alguno de sus módulos a nuestras necesidades,
- o desarrollando e implementando soluciones propias (preferentemente, dada la naturaleza de la herramienta, mediante *scripts*).

Por tanto, es tarea del alumno seleccionar e implementar en cada caso la solución óptima para cada necesidad, así como el proceso previo de análisis de soluciones (y, si cabe, la motivación de la elección mediante comparativas).

El producto final a desarrollar es el siguiente:

- Módulos software integrados en la plataforma de monitorización *Pandora FMS Open Source* que permitan la recogida y explotación de información de distintos dispositivos en red:
 - Equipamiento *WiMAX Alvarion*
 - Equipamiento *Networking Industrial Hirschmann*

- Servidores *Proliant* de la firma *HP*
 - Sistemas de almacenamiento de la firma *EMC*²
 - Soluciones de virtualización de la firma *VMware*
 - SGBD de la firma *Oracle*
 - Servidores de aplicaciones web (*Tomcat*)
- Solución que permita, por un lado, analizar el cumplimiento de los *SLA's* contratados por los clientes; y por otro, generar informes que lo avalen.
 - Documentación final que describa y contextualice el trabajo realizado.

1.2.- Experiencia Previa y Motivación

El fondo teórico más básico de este tipo de herramientas de monitorización se encuentra en "*Introducción a Redes de Computadores*", asignatura obligatoria de 2º curso, y "*Tecnologías e Infraestructuras de Red*", asignatura de 3º curso dentro de la especialidad de Computadores.

En ellas, se tiene una aproximación teórica a los protocolos de red (*IP*, *TCP/IP*, *UDP*, ...), a la arquitectura clásica cliente/servidor o agente/servidor y, en especial, al protocolo *SNMP*. No obstante, podrían nombrarse una gran cantidad de asignaturas que pudieran vincularse (tal vez, de un modo menos directo) a la monitorización en red.

Es evidente que **resulta necesario conocer los entresijos y protocolos subyacentes** de un servicio para poder llevar a cabo una monitorización efectiva del mismo. Por ejemplo, para la monitorización del estado de un sitio web, se debe conocer el protocolo *HTTP* y sus códigos de retorno para poder tratarlos correctamente.

Es por ello que el desarrollo de este proyecto conlleva, de un modo intrínseco, la adquisición de nuevos conocimientos específicos sobre multitud de dispositivos, servicios y protocolos utilizados en entornos reales y empresariales.

En este caso particular, se posee una experiencia previa con herramientas de monitorización de red en un entorno empresarial, dentro del programa de **Prácticas en Empresa** de la titulación.

Durante este periodo, se analizaron y compararon distintas herramientas de monitorización de red, dejando de lado aquellas propietarias y de pago de firmas como *HP* o *Cisco*. Se contemplaron alternativas como *OpenNMS*, *Pandora FMS*, *NetDisco*, *Pandora FMS*, *Big Brother* y demás, aunque finalmente se optó (razonadamente) por una implementación de *Nagios/Centreon*. En total, se llegaron a monitorizar más de 200 equipos y dispositivos en red y más de 600 chequeos específicos.

Es por ello que se tiene un conocimiento práctico del paradigma de funcionamiento de este tipo de herramientas y de los chequeos concretos

que realizan (sobre servidores de *BBDD*, servidores web, servidores de correo, uso de *CPU* y discos, uso de red, ...).

La realización del proyecto sobre *Pandora FMS* supone un enriquecimiento y un complemento ideal para los conocimientos previos sobre monitorización de red. *Nagios* y *Pandora FMS* son, sin lugar a dudas, las herramientas de monitorización de código abierto más utilizadas globalmente, contando con clientes como *Movistar*, *Casio*, *Siemens*, *Sony*, ...

Referencias/Enlaces del Capítulo

***NOTA:** Los enlaces han sido comprobados por última vez el día 05/06/2013. Es posible que algunos de ellos estén rotos o hayan sido actualizados. En tal caso, conviene acceder al directorio raíz o página inicial e iniciar la búsqueda de la referencia.

- *Pandora FMS*:
<http://pandorafms.com/>
- *Nagios*:
<http://www.nagios.com/>
- *Centreon*:
<http://www.centreon.com/>
- *NetDisco*:
<http://www.netdisco.org/>
- *Big Brother*:
<http://bb4.com/>
- *Pandora FMS Customers*:
<http://pandorafms.com/Cientes/Cientes/en>
- *Nagios Customers*:
<http://www.nagios.com/users>

2.- MARCO DE TRABAJO INICIAL

El proyecto se desarrolla en dos entornos de trabajo diferenciados:

- **Puesto de trabajo COM&MEDIA:** Es el puesto de trabajo acomodado para el alumno en las propias instalaciones de la empresa. Cuenta con el equipo indicado a continuación:
 - *HP Compaq dc7600 Convertive Minitower*
 - *CPU Intel Pentium D 3'40 GHz (Dual Core)*
 - *RAM 3 GB*
 - *SO Windows XP Professional 5.1 32 bits*
 - Software instalado de utilidad:
 - *VMware Workstation 9.0.1*
 - *Internet Explorer 8*
 - *Suite ofimática Microsoft Office 2007*
 - *SonicWall NetExtender Windows Client (MSI)*

- **Entorno no presencial,** trabajando sobre el equipo portátil personal:
 - *MacBook Pro 7.1*
 - *CPU Intel Core 2 Duo 2.66 GHz (Dual Core)*
 - *RAM 4 GB*
 - *SO Mac OS X Snow Leopard 10.6.8*
 - Software instalado de utilidad:
 - *VMware Fusion 4.1.3*
 - *Google Chrome 25.0.**
 - *Mozilla Firefox 16.0.2*
 - *Suite ofimática Microsoft Office 2011 for Mac*

En ambos entornos se cuenta con acceso a Internet y con permisos de administración locales.

3.- PRIMERA FASE

En esta primera fase, se analizará, desde distintas perspectivas, la herramienta *Pandora FMS Open Source*, actualmente utilizada por la empresa para la monitorización de redes y sistemas de varios de sus clientes.

Este análisis no forma parte de los productos finales que la empresa espera, sino que servirá de base intelectual para que éstos puedan ser desarrollados.

3.1.- Análisis de *Pandora FMS*

El desarrollo de *Pandora FMS (Flexible Monitoring System)* está liderado por *Ártica Soluciones Tecnológicas*, empresa tecnológica de I+D española que ofrece herramientas basadas en software *OpenSource*.



3.1.1.- Licencias del Uso

Ártica ST ofrece dos productos relacionados con *Pandora*:

- ***Pandora FMS Open Source***, herramienta de monitorización clásica y gratuita, bajo licencia *GPL2*.
- ***Pandora FMS Enterprise***, herramienta de pago que ofrece un mayor número de funcionalidades nativas que la versión *Open Source*, además de soporte técnico y actualizaciones.



Dado que nuestra herramienta de trabajo será exclusivamente la versión de código abierto, en lo sucesivo trataremos indistintamente los términos "*Pandora FMS*" y "*Pandora FMS Open Source*" para referirnos a ella.

3.1.2.- Arquitectura. Elementos de una Instalación Básica.

Una instalación básica de *Pandora FMS* comprende los siguientes elementos:

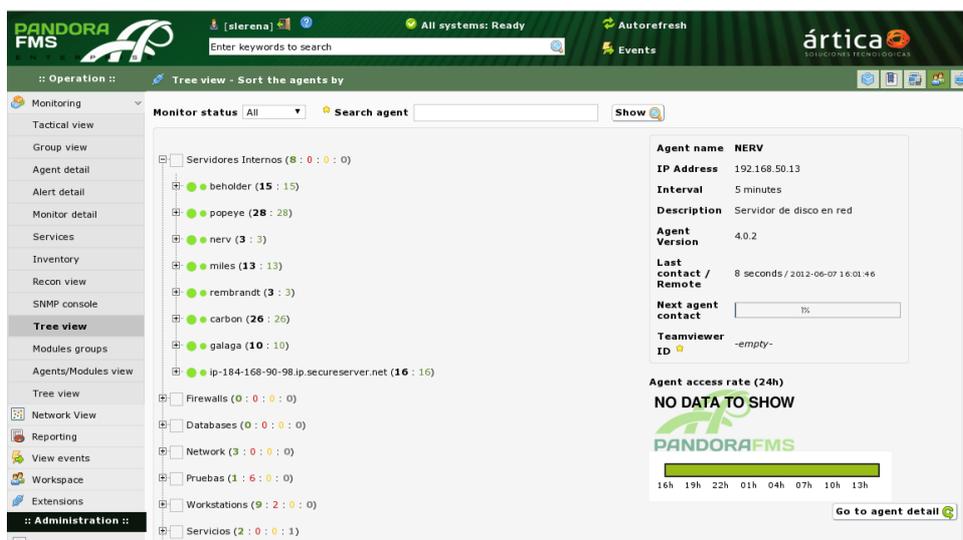
- ***Pandora FMS Server***: es el servidor central de la herramienta, generalmente lanzado en una máquina *Linux (OpenSUSE, Debian, RHEL* o cualquiera de sus respectivos derivados) y localizable en

`/etc/init.d/pandora_server`

o sus equivalentes para otras distribuciones. También puede compilarse su código y lanzarse en otros SO basados en *Unix* como *FreeBSD*, *Mac OS X*, *Solaris*, ...

En la misma máquina, el *Pandora FMS Server* operará junto con los siguientes servicios:

- **Servidor Web**, tradicionalmente *Apache*, que alojará la interfaz web para el uso y administración de la aplicación.



- **Base de Datos**, por defecto *MySQL*, que actuará de *back-end* general de la aplicación. Ofrece soporte experimental para otros *SGBD* como *Oracle* y *PostgreSQL*, aunque no nos resultarán de interés.
- **Tentacle Server**, servidor que da soporte al protocolo denominado "*Tentacle Protocol*", que viene a sustituir otros protocolos de intercambio de ficheros como *SCP* o *FTP*. Responde al puerto *TCP/41121* (asignado por la *IANA*).

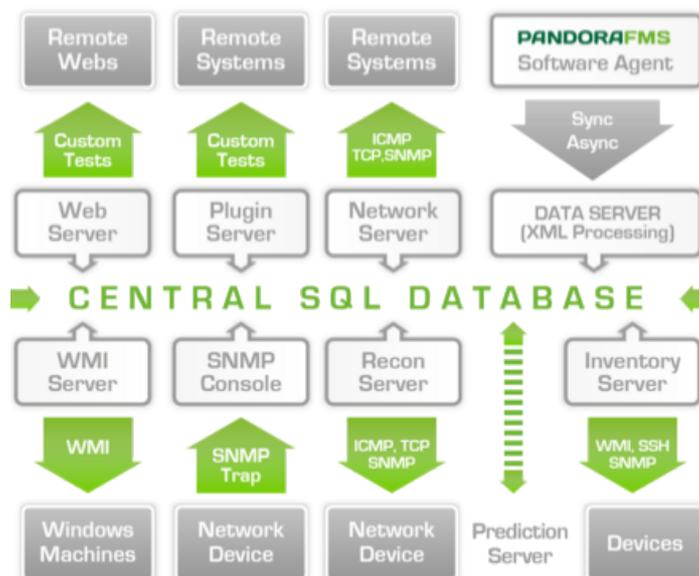


Su función es la de transferir información de forma segura¹ entre un agente (que mencionaremos a continuación) y un servidor *Pandora FMS*. Generalmente se encuentra bajo la ruta:

```
/etc/init.d/tentacle_serverd
```

¹ Se emplea autenticación mediante certificados *X.509* y se cifra la comunicación. Estos parámetros de seguridad pueden variar para distintas instalaciones.

- Agrupamos el resto de servicios bajo los denominados **Servidores de Chequeos Directos** o sin agente software. Se incluyen servicios como la consola *SNMP*, el servidor *WMI*, el servidor de red para envío de paquetes *ICMP*, *TCP* o *SNMP*, el servidor de plugins,...



- Otro elemento básico (ya mencionado) de la aplicación es el **Agente Software**. Un agente es un proceso que se ejecuta en una máquina anfitriona y obtiene datos tales como:
 - Uso de *CPU*
 - Uso de discos
 - Procesos activos
 - Nivel de batería (en casos aplicables)
 - ...

Tras recabar la información, el agente la envía en formato *XML* al *Pandora FMS Server* mediante el mencionado *Tentacle Protocol* para que sea procesada. Existe un agente para cada uno de la mayoría de *SO* tradicionales, así como para *Android* (que recibe el nombre de *Pandroid*) y otros sistemas embebidos.



- Por último, mencionamos aquí al **resto de dispositivos en red** que son objeto de los chequeos de la aplicación, sin agente software. Es la

propia máquina anfitriona del servidor de Pandora la que se encarga de realizar estos chequeos, tales como:

- *Pings*
- *Peticiones HTTP*
- *Consultas SNMP*
- *Captura de traps SNMP*
- *Consultas mediante WMI (Windows Management Instrumentation)*
- ...

Estos dispositivos pueden ser:

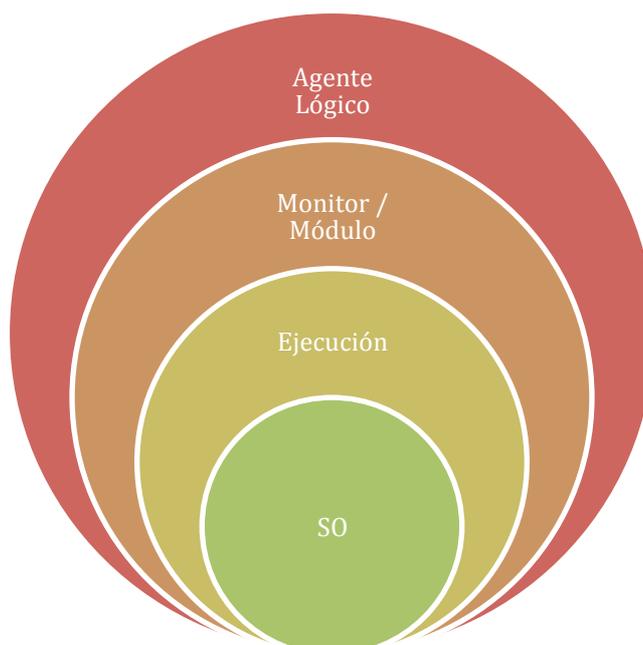
- *Routers*
- *Switches*
- *Firewalls*
- *APs WiFi*
- y, en definitiva, cualquiera que tenga acceso a la red.

Aunque existen otras configuraciones más escalables y redundantes en *HA (High Availability)*, para nuestros fines bastará con considerar una instalación básica.

3.1.3.- Funcionamiento de un Chequeo

3.1.3.1.- Modelo de Capas

En esta sección, describiremos cómo opera la aplicación para realizar un chequeo y procesar la información que este entrega. Prestaremos especial atención al modelo de capas que presenta, analizándolo en orden descendiente:



- El nivel más alto que consideraremos es el del **Agente Lógico**. *Pandora FMS* organiza todos sus chequeos en base a este tipo de agrupaciones, que pueden corresponderse con agentes software instalados en otras máquinas o simplemente servir como agrupación de otros chequeos independientes.

Ilustrándolo en un ejemplo, consideramos una instalación de *Pandora* en la que monitorizamos dos agentes, con los siguientes nombres:

- ServidorWindows1, un servidor *Windows* donde coexisten una serie de módulos, de nombre: *CPU, Disco C:\, Disco D:\, RAM*. La información de estos módulos es extraída mediante un agente software instalado en el propio servidor.
- RouterCisco1, donde operan los siguientes módulos: *Ping, InterfazFE0/0, Temperatura*. La información de estos módulos no se obtiene a través de un agente software, sino que es el mismo servidor de *Pandora FMS* el que realiza los *pings* y consultas *SNMP* pertinentes.

Toda esta información (agentes lógicos, módulos y sus relaciones) se almacenan en la base de datos *MySQL* del servidor central de *Pandora FMS*.

- En el siguiente nivel, como ya hemos adelantado, se encuentran los **Módulos**. En *Pandora*, un módulo es una representación de un chequeo como tal: una consulta *SNMP*, un *ping*, un dato concreto ofrecido por un agente software, etc.

Un módulo asociado a un estado recibe el nombre de **Monitor**². El estado de un monitor puede ser:

- *Critical* (Color rojo)
- *Warning* (Color amarillo)
- *OK* (Color verde)
- *Unknown* (Color gris)

Por tanto, en este nivel se determinan los rangos de valores para los cuales se produce un estado determinado. Por ejemplo, en el caso del módulo *CPU* del servidor *Windows* descrito anteriormente podríamos delimitar los estados de la siguiente manera, en función del uso de la *CPU*:

- 0 - 70% -> *OK*
- 70 - 90 % -> *Warning*
- 90 - 100% -> *Critical*

² En la comunidad de usuarios, e incluso en la propia documentación oficial, es habitual utilizar los términos Módulo y Monitor indistintamente. En este documento, tampoco seremos estrictos con los matices que presentan formalmente.

Además, *Pandora FMS* permite definir qué tipo de dato (texto, entero, booleano, ...) utilizará el módulo para definir su estado.

- Todo módulo lleva asociada la **Ejecución** de un comando *shell* o un *script* personalizado. Esta será la acción que el módulo realizará periódicamente (cada X tiempo, indicado en la configuración del módulo) y la que proveerá al módulo de información que, posteriormente, debe tratar para definir su propio estado y generar las alertas oportunas.

Este es el nivel más bajo que consideraremos en la aplicación, ya que colinda con el intérprete de comandos del propio sistema operativo.

3.1.4.- Desarrollo de *Plug-Ins*

3.1.4.1.- Dentro de un *Agente Software*

La aplicación nos permite crear nuestros propios ficheros ejecutables (a nivel de Ejecución) para realizar los chequeos y las acciones que deseemos.

Estos ejecutables deben cumplir las siguientes restricciones:

- Puede estar escrito en cualquier lenguaje de *scripting* o compilado, siempre y cuando pueda invocarse como fichero ejecutable desde un *shell* o *cmd* (dependiendo de la máquina en la que se encuentre instalado el agente software).

Generalmente, en sistemas *UNIX/Linux*, estos *scripts* se almacenan bajo el directorio:

```
/usr/share/pandora_agent/plugins/
```

Y en sistemas *Windows*:

```
%ProgramFiles%\Pandora_Agent\util\
```

- Debe ser completamente automático, sin interacción alguna por parte del usuario.
- Debe entregar información por la salida estándar (*echo*, *printf* o equivalentes), en formato *XML* y utilizando la siguiente sintaxis:

```
<module>
<name><![CDATA[nombre_del_módulo]]></name>
<type><![CDATA[tipo_de_datos]]></type>
<data><![CDATA[datos_devueltos]]></data>
<description><![CDATA[info_adic]]></description>
</module>
```

Para que opere, entro del fichero de configuración del agente software, se indica la siguiente directiva:

```
module_plugin ruta_del_ejecutable
```

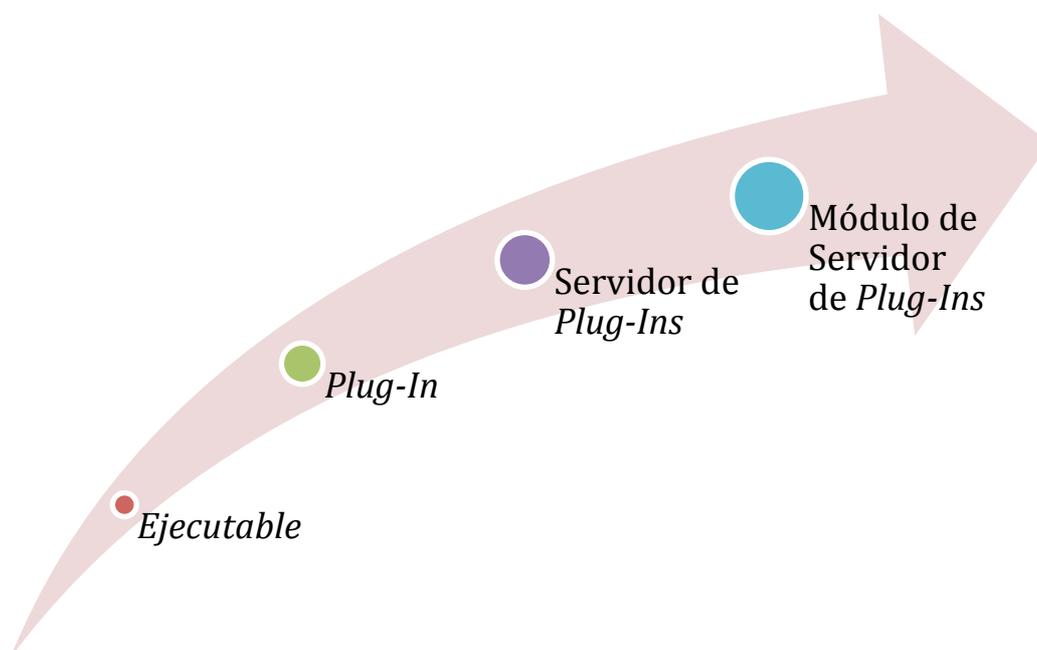
De este modo, *Pandora FMS* ofrece una gran libertad al permitir monitorizar cualquier característica que podamos acceder a través de la línea de comandos.

Además, *Pandora* ofrece un *wrapper* para *plug-ins* desarrollados para la plataforma de monitorización *Nagios*, lo que permite reutilizarlos indicando la directiva:

```
module_plugin nagios_plugin_wrapper ruta_del_ejecutable
```

3.1.4.2.- Dentro del Servidor de Plug-Ins

El propio servidor de *plug-ins* de la aplicación nos permite crear nuestros propios *scripts* para realizar chequeos remotos (por lo que las posibilidades de monitorización son muy inferiores a las que ofrece un *plug-in* de un agente software instalado en la propia máquina). El flujo de la información es el siguiente:



- El **Ejecutable**, de forma análoga al caso anterior, deberá cumplir las siguientes restricciones:
 - Puede estar escrito en cualquier lenguaje de *scripting* o compilado, siempre y cuando pueda invocarse (como fichero ejecutable) desde un *shell*.

Generalmente, estos *scripts* se almacenan bajo el directorio:

```
/usr/share/pandora_server/util/plugin/
```

- Debe ser completamente automático, sin interacción alguna por parte del usuario.
- Debe entregar información por la salida estándar.
- A través de la interfaz web, podemos crear un **Plug-In**, indicando parámetros como:
 - Nombre
 - Ruta del *script* a ejecutar
 - Parámetros que acepta el *script* y cómo se indican ("-i" para indicar una dirección IP, "-p" para indicar un número de puerto, ...).
 - ...
- El **Servidor de Plug-Ins** no es más que una agrupación que abarca los *plug-ins* descritos anteriormente.
- Un **Módulo de Servidor de Plug-Ins** es aquel módulo que recoge y trata la información provista por un *Plug-In* determinado. A este nivel se indican los datos como la dirección IP y número de puerto mencionados en el ejemplo anterior.

3.1.5.- Generación de Alertas

Pandora FMS es capaz de procesar la información recolectada y lanzar una serie de disparadores que hayan sido configurados. Siguiendo su propia nomenclatura (la cual puede que no resulte demasiado intuitiva), la herramienta estructura la generación de alertas de este modo:



- **Command** (Comando) es el *script* que se ejecutará cuando se dispare la alerta. Puede consistir en enviar un correo electrónico, enviar un SMS, generar un *trap SNMP*, ejecutar otro *script* concreto dentro de la misma máquina, etc.

Las posibilidades son prácticamente ilimitadas, pero, por supuesto, para poder realizar estas acciones se debe preparar y configurar la máquina anfitriona según corresponda (acceso y permisos sobre un servidor de correo, acceso a la red telefónica, ...)

- **Action** (Acción) definirá los parámetros concretos de un comando. Dentro del ejemplo de una alerta por *email*, podría indicarse, entre otras cosas, quién será el destinatario de la alerta y cuál será el asunto del correo a enviar.
- La configuración del **Template** o Plantilla consta de tres apartados:
 - **Condiciones**, donde se indicará en qué estado debe estar un módulo para que se dispare la alerta (*critical, warning, ...*).
 - **Firing** (Disparo), donde se indican otras condiciones que deben ser satisfechas para que la alerta se dispare.

Por ejemplo, podría configurarse de tal modo que una alerta solo pudiera dispararse "de lunes a viernes, entre las 09:00 y las 14:00".

En este apartado se indica también la *acción* por defecto de la plantilla (como el envío de un *email*) y el mensaje a enviar (utilizando una serie de macros que Pandora ofrece).

- **Recovery** (Recuperación), donde se indican las acciones a realizar cuando el módulo asociado se recupere de su estado de error.

Podría interesarnos, por ejemplo, que al monitorizar el estado de una línea de comunicación se nos alertara en el momento en el que se cae la línea y en el momento en el que vuelve a estar operativa.

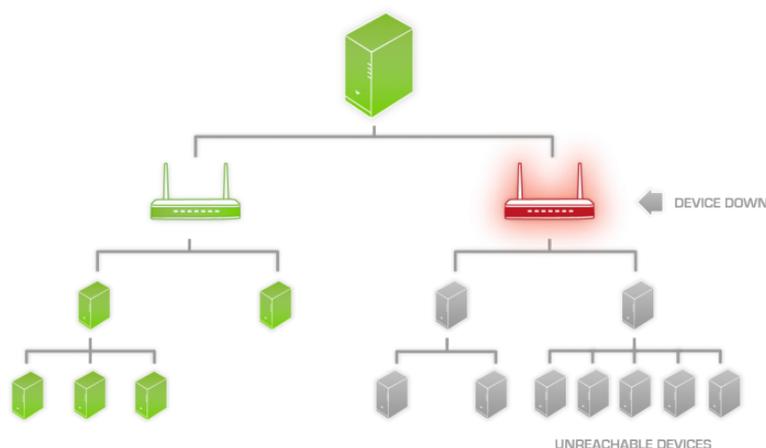
PARAMETERS CARRYING



Para finalizar, se debe **enlazar el *template* con el módulo** deseado (un *ping*, una consulta *SNMP*, ...). De este modo, cada vez que el módulo entre en un estado determinado, se ejecutarán las alertas o disparadores configurados.

3.1.5.1.- Protección en Cascada

La protección en cascada o **Cascade Protection** es una funcionalidad opcional de *Pandora FMS* que permite detectar el fallo de un *router*, *switch* o equipo del que dependan otros dispositivos, para alertar únicamente de la caída de éste y omitir la del resto de dispositivos inalcanzables.



Esta protección evita que se generen ingentes cantidades de alertas por el fallo de un único dispositivo. Para ello, las dependencias entre agentes lógicos (relaciones padre/hijo) deben de estar debidamente definidas en el sistema.

3.1.6.- Reporting

En este apartado se analizará la capacidad nativa de *Pandora FMS* para generar gráficos e informes de la monitorización de módulos en un periodo de tiempo determinado.

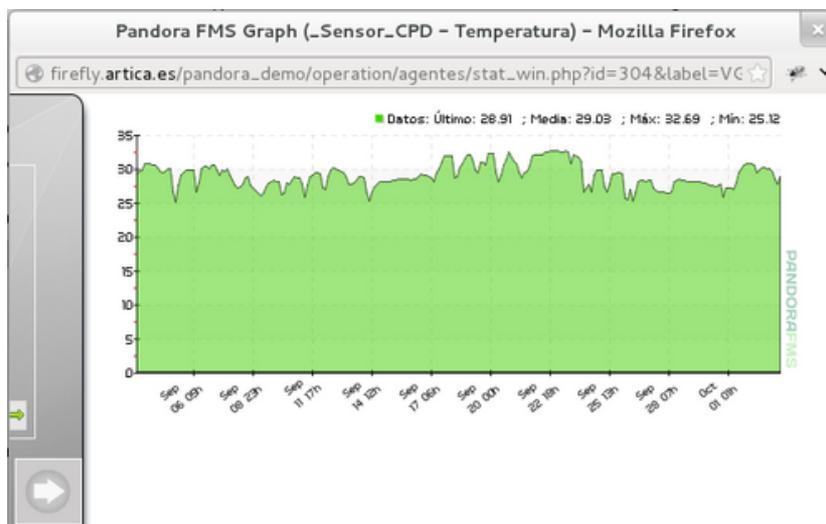
Cabe mencionar que en la tercera fase de este proyecto ([5.- TERCERA FASE](#)) se profundiza en este aspecto, ya que cubre la solución exigida por la empresa de *reporting* y de análisis de cumplimiento de *SLAs*.

3.1.6.1.- Generación de Gráficos

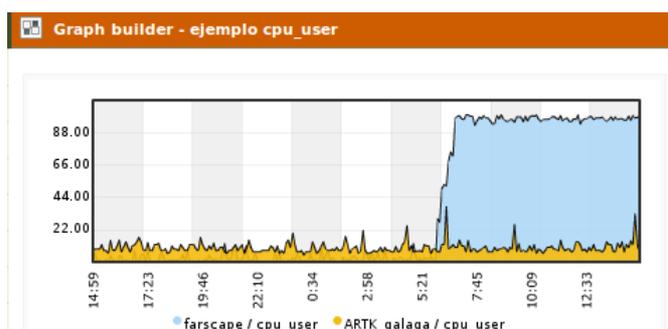
Pandora FMS diferencia entre dos tipos de gráficos::

- Los **gráficos simples**, que no son más que gráficos que *Pandora* genera automáticamente a partir de información de módulos. Todo módulo operativo puede ser representado en uno de estos gráficos, diferentes en función del tipo de datos que empleen.

Los gráficos pueden generarse en formato *Flash* o *PNG*. No se almacenan en el sistema, sino que la propia aplicación los genera en cada petición:



- **Gráficos combinados**, que pueden integrar información de varios módulos conjuntamente:



3.1.6.2.- Generación de Informes

Pandora FMS posee una potente infraestructura para generar informes, con multitud de modelos de informes predefinidos. A grandes rasgos, es posible crear informes que, para un determinado periodo de tiempo, incluyan:

- **Valores máximos, mínimos o medios** de un módulo.
- **Gráficos** (los mencionados en [3.1.5.1.- Generación de Gráficos](#)).
- **Alertas disparadas** por un módulo, agente o grupo de agentes.
- **Eventos generados** en el sistema por un módulo, agente o grupo de agentes.
- **Número de agentes en función de su estado**, entre un grupo de agentes o entre el sistema completo.

- **MTBF** (*Mean Time Between Failure*) o tiempo medio que transcurre entre fallos de un módulo. Es un indicador muy útil de la fiabilidad de un sistema.
- **MTTR** (*Mean Time To Recovery*) o tiempo medio entre el fallo y la recuperación de un módulo.
- **Análisis de un SLA** (*Service Level Agreement*), que resulta un tipo de informe más complejo que el resto. En este caso, se indican los siguientes parámetros para cada módulo afectado:
 - **Valor mínimo** que debe cumplir.
 - **Valor máximo** que no debe rebasar.
 - **Límite (%)**. El *SLA* se mostrará como correcto si el valor del módulo se ha mantenido entre el máximo y el mínimo durante el porcentaje de tiempo estipulado, y se mostrará como incorrecto en caso contrario.

En el caso de los *SLA* que combinen varios módulos, su nivel de cumplimiento dependerá del nivel de cumplimiento de cada uno de los *SLAs* definidos.

- Etcétera.

Estos informes pueden ser exportados en *HTML*, *XML*, *CSV* o *PDF* (en la versión *Enterprise*). Asimismo, pueden configurarse para ser enviados periódica y automáticamente por correo electrónico.

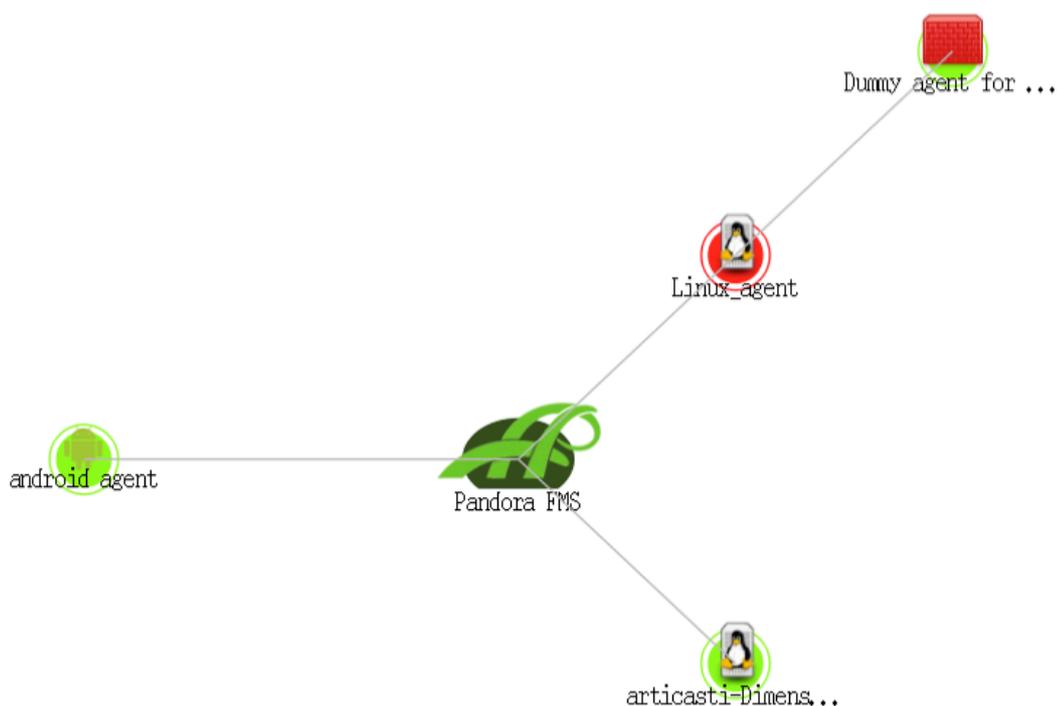
3.1.7.- Mapas de Red

La representación gráfica de los agentes en un mapa de red es una característica que se ve más favorecida en la versión *Enterprise*, lo que significa que la capacidad de la versión *Open Source* en este aspecto resulta algo pobre.

Puesto que el desarrollo del proyecto no incluye la explotación de esta funcionalidad, nos limitamos a mostrar el aspecto de este tipo de mapas topológicos.

Cada nodo representa a un agente, y el color de la base representa el estado de dicho agente³:

³ Recordemos que un agente lógico adquiere el peor de los estados de los módulos que agrupe. Así, si un agente tiene un módulo en estado *critical*, otro en *warning* y otro en *OK*, el estado del agente será *critical*.



3.1.8.- Gestión de Usuarios y Permisos

Por defecto, *Pandora FMS* utiliza su propio sistema de gestión de usuarios, almacenados en su propia base de datos. Sin embargo, es posible y habitual que las organizaciones integren *LDAP* o *AD* para unificar el proceso de autenticación.

La aplicación gestiona los permisos de los usuarios mediante la definición de *ACL (Access Control List)*, donde se definen distintos perfiles de usuario con distintos permisos de acceso o modificación a los recursos de la aplicación.

Dado que no reviste mayor importancia dentro del ámbito del proyecto, no profundizaremos más en el análisis de este apartado.

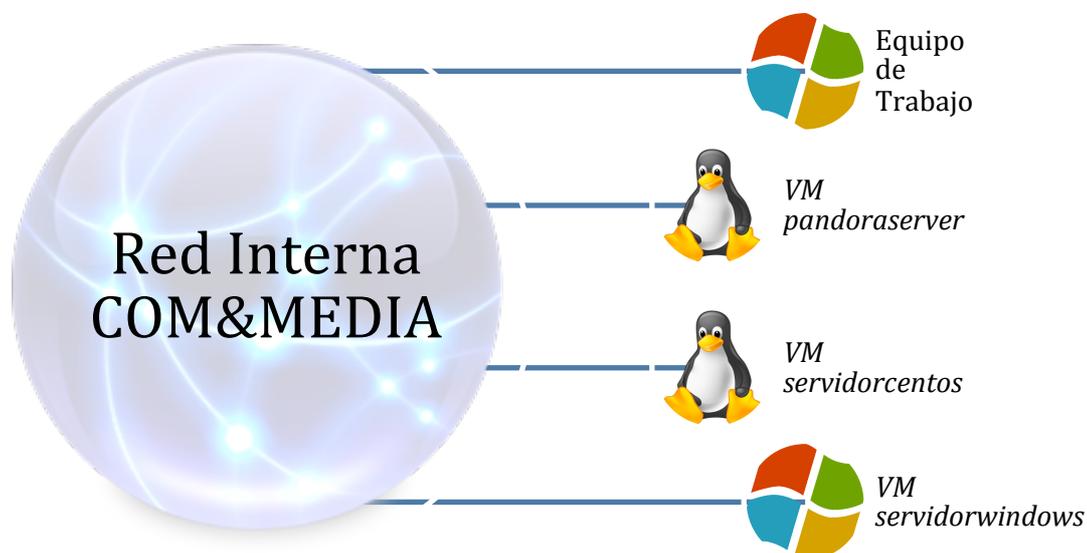
3.2.- Instalación de *Pandora FMS* en un Entorno Virtual

Esta sección cubre la instalación y puesta en marcha de *Pandora FMS* sobre una máquina virtual, de modo que sirva de base para el futuro desarrollo de soluciones.

La razón de utilizar un entorno virtual es no afectar a la continuidad de negocio de la empresa y no poner en riesgo otras instalaciones de *Pandora FMS* operativas. Además, el hecho de trabajar con una instalación limpia implica un aprendizaje más profundo que trabajar con una plataforma ya configurada.

El entorno virtual contará con varias *VM* instaladas en red, para poder emular un entorno de monitorización, que será descrito en los siguientes subapartados.

Dado que las interfaces de acceso a red se configurarán en modo *bridge* (puenteado, con acceso directo a la red mediante *DHCP* o dirección *IP* estática), el entorno resultante debería seguir el esquema:



Esta sección no pretende ser un manual de instalación, sino un informe de la instalación y configuración de nuestra propia instancia de *Pandora FMS*.

3.2.1.- VM de Pandora FMS Server

Ajustándonos a los estándares de la empresa, el servidor central de *Pandora FMS* se ejecutará sobre una máquina virtual *Linux CentOS 6.4 i686*. Es una instalación mínima (sin interfaz gráfica alguna) a la que le hemos asignado los siguientes recursos:

- Nombre de máquina: *pandoraserver*
- *HDD* de 20 GB
- *RAM* de 1 GB
- 1 núcleo de *CPU*

3.2.1.1.- Instalación de Dependencias

Dado que se ha escogido una instalación del tipo *minimal* (que únicamente contiene los elementos necesarios para el funcionamiento de la máquina, sin ningún paquete adicional), se han instalado las siguientes dependencias principales de *Pandora FMS*, además de ciertos módulos asociados a sus respectivos:

- Intérprete de *Perl*.
- Intérprete de *PHP*.
- Servidor *MySQL*.
- Servidor *Apache*.
- Servidor *Tentacle*.
- ...

Todo ello a través de los gestores de paquetes de línea de comandos de *CentOS* y *RHEL*:

- *yum*, para el cual se han actualizado y añadido nuevos repositorios.
- *rpm*, con el cual se han instalado paquetes con extensión ".rpm" previamente descargados en la máquina (principalmente, mediante el comando *wget*).

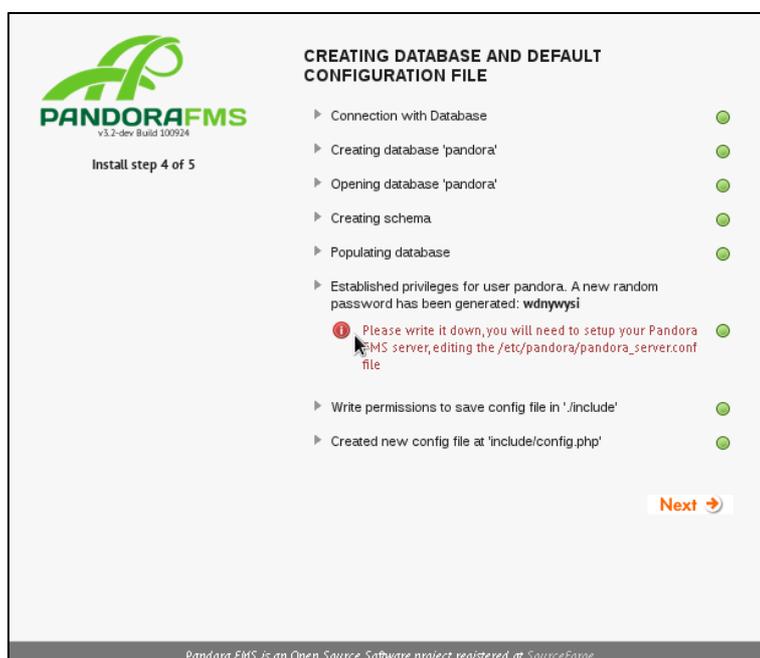
3.2.1.2.- Instalación de Pandora FMS Server

Una vez arrancados los servicios *mysqld*, *httpd* y *tentacle_serverd* (servidor de base de datos, web y *tentacle*, respectivamente), se han descargado e instalado los paquetes correspondientes al agente software, servidor y consola de *Pandora FMS*.

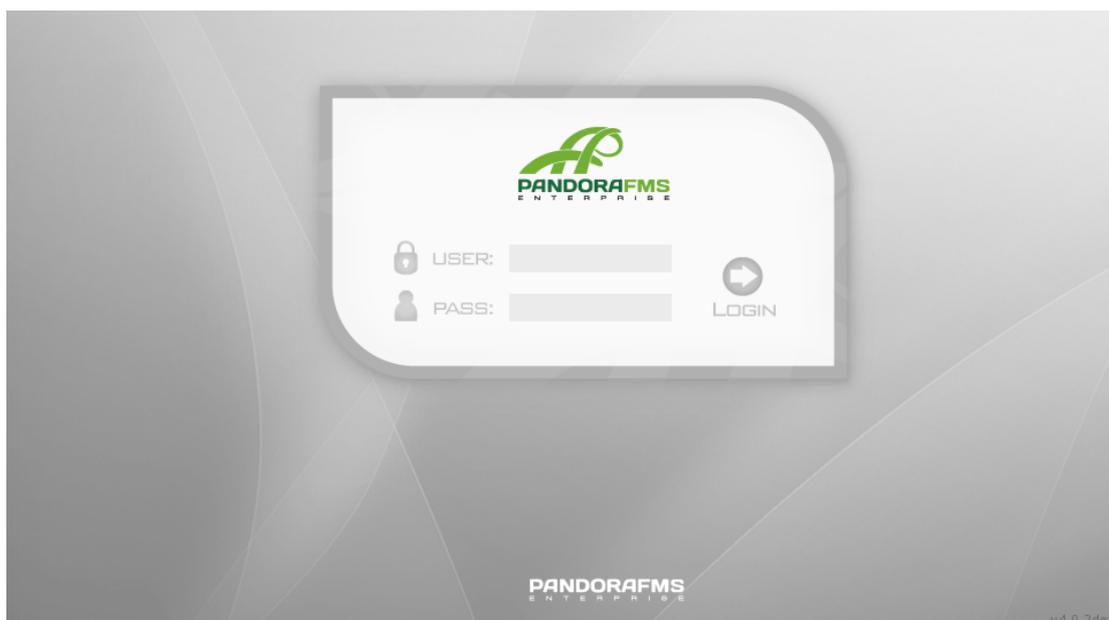
Para proseguir con la instalación se nos insta a emplear un instalador web, lo que hace necesario acceder desde el navegador web de una máquina externa (con acceso por red a la VM, evidentemente) a la siguiente URL:

http://dir_IP_Pandora_FMS_Server/pandora_console/install.php

Con ello lanzaremos el *wizard* de instalación del servidor de *Pandora*, que nos ayudará a configurar la base de datos que utilizará como *backend*, chequeará el cumplimiento de dependencias, etc:



Una vez finalizado, se mostrará la pantalla de *login*, a través de la que finalmente podremos acceder a la aplicación:



***NOTA:** Por comodidad y para favorecer la resistencia de la aplicación ante cortes eléctricos o reinicios inesperados, se ha programado el autoarranque de los servicios *httpd*, *mysqld*, *snmpd*, *tentacle_serverd*, *pandora_agent_daemon* y *pandora_server* mediante la utilidad *chkconfig*.

3.2.1.3.- Configuración de Email

En esta sección, configuraremos el servidor *CentOS* para que sea capaz de enviar correos electrónicos vía *SMTP*.

Para ello, en primer lugar, creamos la siguiente cuenta de correo *GMail*:

alertas.pandora.fms@gmail.com

Modificamos el fichero de configuración de *Pandora FMS Server* en:

```
/etc/pandora/pandora_server.conf
```

Y editamos las líneas correspondientes al *MTA (Message Transfer Agent)*, de modo que la aplicación utilice la cuenta *GMail* recientemente creada para enviar alertas *email*.

Dado que *GMail* requiere autenticación *SSL/TLS* frente a su servidor *SMTP* (no soportado por defecto en *Pandora*), debemos configurar el servicio *Postfix* para habilitar el procedimiento de autenticación mencionado.

Desde la interfaz web de *Pandora FMS*, definiremos una acción de alerta que envíe un email (a una dirección de correo personal) cuando un determinado módulo entre en estado *critical* o, estando en estado *critical*, vuelva a su estado normal.

3.2.1.4.- Configuración de Servidor de Recepción de Traps SNMP

Para que *Pandora FMS* pueda recibir y tramitar *traps SNMP*, realizamos una pequeña modificación del fichero de configuración del servicio *snmptrapd*,

indicando que tramite aquellos *traps* que pertenezcan a la comunidad "*public*".

3.2.2.- VM de Servidor Linux

De forma muy similar a la máquina virtual anterior, generamos otra máquina virtual *Linux CentOS 6.4 i686*, con el único objetivo de instalar un agente de *Pandora FMS* en ella. Por tanto, los recursos asignados son más reducidos:

- Nombre de máquina: *servidorcentos*
- *HDD* de 10 GB
- *RAM* de 512 MB
- 1 núcleo de *CPU*

3.2.2.1.- Instalación y Configuración de Agente

Comenzamos descargando el paquete ".rpm" correspondiente al agente software e instalándolo mediante la utilidad *rpm*.

Las dependencias necesarias han sido instaladas a mediante el gestor de paquetes *yum*, aunque, en este caso, apenas existen. Para el funcionamiento del agente no será necesario instalar ningún tipo de servidor aparte del *daemon* del propio agente software.

Accederemos al fichero de configuración del agente para configurarlo (indicando la @IP del servidor central, básicamente). Dicho fichero puede hallarse en la ruta:

```
/etc/pandora/pandora_agent.conf
```

Tras reiniciar el servicio asociado al agente (*pandora_agent_daemon*), observamos que ha sido capaz de interactuar con el servidor central, ya que la interfaz web de la aplicación muestra el agente software recién creado (*servidorcentos*) y 10 módulos operativos y generados por defecto:

F.	Type	Module name	Description	Status	Warn	Data	Graph	Last contact
		/dev/mapper/VolGroup-lv_root	% of usage in this volume		N/A - N/A	10		1 minutes 12 seconds
		/dev/sda1	% of usage in this volume		N/A - N/A	11		1 minutes 13 seconds
		Cache mem free	Free cache memory in MB		N/A - N/A	443		1 minutes 13 seconds
		cpu_user	User CPU Usage (%)		N/A - N/A	0		1 minutes 13 seconds
		Cron task files	Number of cron task files		N/A - N/A	2		1 minutes 13 seconds
		LastLogin	Monitor last user login		N/A - N/A	root pts		1 minutes 13 seconds
		Load Average	Average process in CPU (Last minute)		N/A - N/A	0		1 minutes 13 seconds
		proctotal	Total processes		N/A - N/A	77		1 minutes 13 seconds
		sshDaemon	Check ssh service		N/A - N/A	2		1 minutes 13 seconds
		tmpfs	% of usage in this volume		N/A - N/A	0		1 minutes 13 seconds

3.2.3.- VM de Servidor Windows

De nuevo, preparamos otra máquina virtual *Windows Server 2003 Standard Edition* de 32 bits, de modo que resulte ligera para que la máquina física que aloja todas las VMs pueda trabajar con fluidez. Los recursos asignados a esta máquina virtual son:

- Nombre de máquina: *servidorcentos*
- *HDD* de 10 GB
- *RAM* de 512 MB
- 1 núcleo de *CPU*

3.2.3.1.- *Instalación y Configuración de Agente*

Aprovechando que esta máquina virtual posee un entorno gráfico, instalamos el agente software a través de su propio asistente gráfico (obtenido en la sección de descargas del *website* oficial de *Pandora FMS*).

Observamos que, tras indicar la dirección *IP* de nuestro servidor central de *Pandora*, el agente recién instalado es capaz de interactuar con el propio servidor para generar automáticamente un agente lógico en *Pandora FMS*, con 5 módulos por defecto.

Referencias/Enlaces del Capítulo

***NOTA:** Los enlaces han sido comprobados por última vez el día 05/06/2013. Es posible que algunos de ellos estén rotos o hayan sido actualizados. En tal caso, conviene acceder al directorio raíz o página inicial e iniciar la búsqueda de la referencia.

- *Pandora FMS Open Source vs Enterprise:*
[http://pandorafms.com/Community/Enterprise vs. Open Source /en](http://pandorafms.com/Community/Enterprise%20vs.%20Open%20Source/)
- *Pandora FMS Open Source:*
http://pandorafms.com/Pandora/Pandora_FMS_Opensource/en
- *Pandora FMS Enterprise:*
[http://pandorafms.com/Pandora/Pandora FMS Enterprise/en](http://pandorafms.com/Pandora/Pandora_FMS_Enterprise/en)
- *Pandora FMS Video Tutorials:*
http://pandorafms.com/Community/video_tutorial/en
- *Tentacle Server:*
<http://tentacled.sourceforge.net/>
- *Pandora FMS Downloads:*
<http://pandorafms.com/Community/download/en>
- *Pandora FMS Documentation:*
http://www.openideas.info/wiki/index.php?title=Pandora:Documentation_en

4.- SEGUNDA FASE

Habiéndonos sumergido en la herramienta y una vez preparado el entorno virtual, nos disponemos a diseñar e implementar soluciones de monitorización para los distintos dispositivos y sistemas que presenta la empresa.

El procedimiento estándar para lidiar con cada elemento es el siguiente:

- 1) **Análisis del dispositivo o sistema.** Se analizará la arquitectura, funcionamiento y demás características reseñables del elemento objetivo. Asimismo, se definirán los aspectos más interesantes para ser monitorizados (además de los requeridos por la empresa, evidentemente), que posteriormente serán analizados y seleccionados en función de su viabilidad e importancia.
- 2) Analizar los **dispositivos o sistemas de prueba** concretos. La empresa nos otorgará acceso (generalmente, con perfiles *read-only*) a un determinado número de ellos.
- 3) **Creación de agentes** correspondientes a los dispositivos de prueba. Por eficiencia (respecto al consumo de recursos informáticos), no todos los dispositivos de prueba tendrán necesariamente asociado un agente lógico en *Pandora FMS*. No obstante, se emplearán tantos como sean necesarios para poder desarrollar los módulos y evaluar su funcionamiento adecuadamente.
- 4) **Desarrollo de módulos** asociados a dichos aspectos. El código y otras directivas de configuración se incluirán como anexo de este mismo documento. Se procurará razonar las métricas y umbrales seleccionados o seguir las recomendaciones de un tercero reconocido o de la propia empresa.
- 5) **Configuración de alertas**, si se consideraran interesantes y aportaran una solución distinta a las alertas ya configuradas en otros agentes.
- 6) **Evaluación de las soluciones** adoptadas, comprobando su funcionamiento sobre *Pandora FMS* y forzando transiciones entre estados de los módulos (para observar la respuesta del sistema), siempre que sea viable.

Comenzamos a aplicar el procedimiento sobre los dispositivos y sistemas listados por la empresa:

4.1.- Equipamiento *WiMax Alvarion*

4.1.1.- Análisis de Dispositivos

La empresa contempla dos dispositivos concretos:

4.1.1.1.- *Alvarion BreezeNET B*

El equipamiento *Alvarion BreezeNET B* es una solución para establecer una conexión punto a punto a una distancia de hasta 60km, mediante un enlace de radio *OFDM (Orthogonal Frequency Division Multiplexing)*.



Dependiendo del modelo, puede alcanzar velocidades de transmisión desde los 10 Mbit/s hasta los 70 Mbit/s.

La instalación completa cuenta con cuatro dispositivos, que vienen a componer dos estructuras idénticas en ambos extremos de la comunicación.



Cada extremo de la comunicación se compone de los siguientes elementos:

- Unidad de Exteriores (**ODU**), que es la antena que se comunica vía radio con el otro extremo del enlace, que será otra unidad **ODU**. Esta conexión solo soporta cifrado *WEP128* o *AES128*.
- Unidad de Interiores (**IDU**), que interconectará el **ODU** con la red interna (ya sea un *switch*, un *router* de acceso a internet, ...) mediante un cable *1000BaseT (Gigabit Ethernet, GE)*. Su cometido es el de

suministrar energía eléctrica al *ODU* a través de *PoE (Power over Ethernet)*.

Asimismo, cada estación (*ODU + IDU*) posee una denominación especial:

- **BU** (*Base Unit*), que es la estación que posee capacidad de control sobre la estructura.
- La estación contraria, denominada **RB** (*Remote Bridge*), carece de inteligencia y simplemente se conectará a la *BU* para establecer la conexión punto a punto.

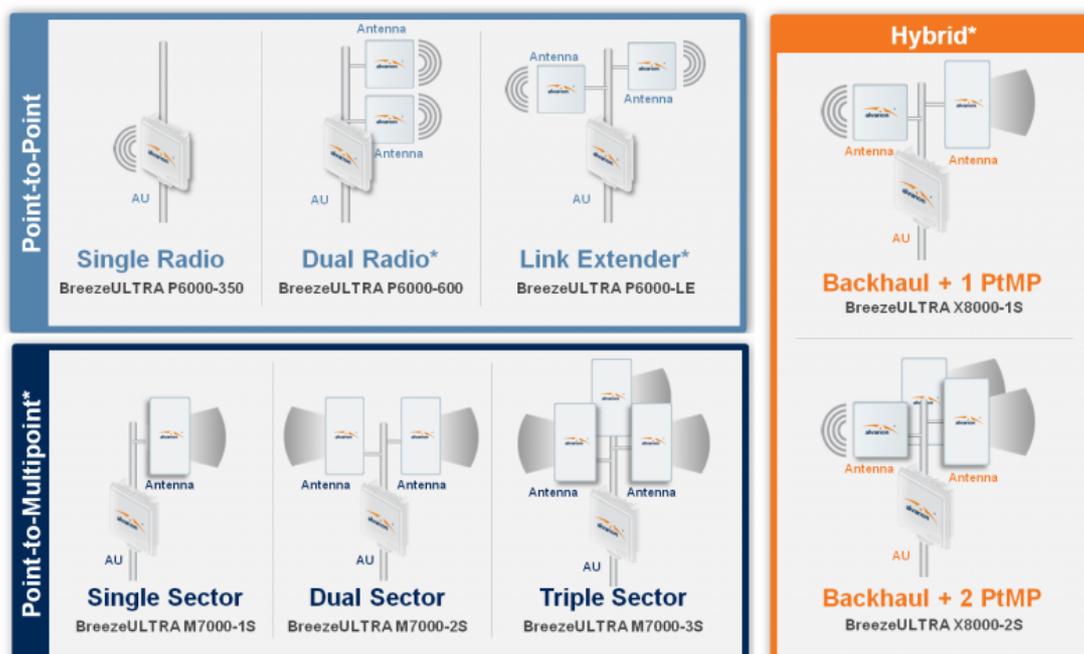
La estructura responde a *SNMP v1*, la versión más antigua de este protocolo, conteniendo los *MIBs* estándares *MIB-2* y *BRIDGE-MIB* y otro particular llamado *Private BreezeACCESS VL MIB*.

Conocido esto, consideramos interesante la monitorización de los siguientes elementos sin agente software:

- Conectividad de cada dispositivo.
- Estado de las interfaces físicas (conectadas, caídas, ...) de la estructura completa.
- Consumo del ancho de banda y pérdida de paquetes en el enlace de radio.
- Nivel de señal de radio, nivel de ruido, ...

4.1.1.2.- Alvarion BreezeULTRA

Este novedoso equipamiento, lanzado al mercado en el año 2012, constituye una versión más completa del *Alvarion BreezeNET* recién analizado, ofreciendo conexiones de punto a multipunto (modelo *M7000*) y punto a punto (modelo *P6000*), entre otras.



Su velocidad de transmisión máxima teórica es de 300 Mbits/s, con un alcance entre los 50 y 120 km, en función de la ganancia de la antena instalada.

Observamos una estructura similar al anterior dispositivo, donde cada estructura se compone de:

- Una **ODU**, que puede ser un único dispositivo con antena integrada (modelo *B350-5X-P6000*) o una unidad de red conectada a una antena externa (modelo *B350D-5X-P6000*). Estas unidades se conectarán entre sí mediante un enlace de radio idéntico al anterior, protegido mediante protocolos y algoritmos más robustos (*WPA2 - AES 128*).
- Una **IDU** que, de nuevo, implementa la tecnología *PoE* sobre *GE*, con lo que consigue alimentar la *ODU* sin necesidad de cables adicionales de alimentación. Cuenta con 2 interfaces físicas *PoE*, de nombre *PoE In* y *PoE Out*.

De nuevo, encontramos la distinción *BU* (Unidad de Base) y *RB* (Puente Remoto).

Dada la escasa información disponible a través de *Internet*, disponemos de los ficheros *MIB* suministrados por el mismo fabricante, cuyas estructuras resultan diametralmente distintas de las estándares de otros dispositivos de red, e incluso, de los modelos *BreezeNET*. En este caso, las consultas se realizarán a través de *SNMP v2*. Además, este modelo ofrece una interfaz web para su configuración y monitorización, la cual puede resultarnos útil en apartados posteriores.

Por lo tanto, nos interesará monitorizar (externamente, sin instalar un agente software):

- Conectividad de cada dispositivo (en función de la instalación concreta).
- Estado de cada interfaz física (conectadas, caídas, ...) de toda la instalación.
- Consumo del ancho de banda y pérdida de paquetes de el/los enlace(s) de radio.
- Nivel de señal de radio, nivel de ruido, ...
- Disponibilidad de la interfaz web de configuración.

4.1.2.- Dispositivos de Prueba

Se nos permite el acceso a la red en la que se encuentran ciertos equipos operativos, a través de VPN⁴.

Estos equipos, agrupados en pares *BU/RB*, recibirán los siguientes identificadores:

Equipos *BreezeNET B100*:

- *BreezeNET_BU1, BreezeNET_RB1*
- *BreezeNET_BU2, BreezeNET_RB2*
- *BreezeNET_BU3, BreezeNET_RB3*

Equipos *BreezeULTRA*:

- *BreezeULTRA_BU1, BreezeULTRA_RB1*
- *BreezeULTRA_BU2, BreezeULTRA_RB2*

4.1.3.- Creación de Agentes

Desde la interfaz web de *Pandora FMS*, procedemos a crear los agentes lógicos asociados a los equipos *BreezeNET_BU1, BreezeNET_RB1, BreezeULTRA_BU1* y *BreezeULTRA_RB1* definidos anteriormente.

4.1.4.- Desarrollo de Módulos

Comenzaremos desarrollando todos los módulos sobre los agente *BreezeNET_BU1* y *BreezeULTRA_BU1*, para posteriormente copiarlos a sus equivalentes *RB*.

Cabe mencionar que actualmente la empresa está monitorizando algunas antenas *BreezeNET*, de modo que tendremos ciertas indicaciones acerca de valores interesantes que monitorizar. Este modelo está siendo paulatinamente sustituido por los nuevos *BreezeULTRA* que se organizan internamente de forma muy diferente a sus predecesores, por lo que tendremos que desarrollar otros módulos (posiblemente muy distintos).

⁴ Para ello, se ha instalado el cliente *SonicWall NetExtender* (software propietario) desde la línea de comandos del *Pandora FMS Server*. Por comodidad, también se ha configurado la máquina física de la empresa para acceder a la red VPN.

Se ha utilizado el software *iReasoning MIB Browser* para navegar por las distintas ramas de los *MIB* y localizar los *OID* más interesantes. Establecemos una periodicidad de chequeo de 5 minutos por defecto.

4.1.4.1- Módulos *BreezeNET*

Creamos los siguientes módulos:

- **Host Alive:** un simple *ping*, lanzado cada 2 minutos.
- **Enlace de Radio (Estado):** Monitorización del valor del *OID* .1.3.6.1.2.1.2.2.1.8.1 (*ifOperStatus*). Cualquier valor distinto de 1 (*Up*) se considera erróneo (2 *Down*, 3 *Testing*, 4 *Unknown*, 5 *Dormant*, 6 *NotPresent*, 7 *LowerLayerDown*). La frecuencia de chequeo es de 3 minutos.
- **Enlace Ethernet (Estado):** Monitorización del valor del *OID* .1.3.6.1.2.1.2.2.1.8.2 (*ifOperStatus*), del mismo modo que el enlace de radio.
- **Enlace de Radio (Bajada):** *Throughput* (bits/s) de bajada en el enlace de radio. *OID* .1.3.6.1.2.1.2.2.1.10.1. La frecuencia de chequeo es de 3 minutos.

Para calcular este valor, se obtiene la diferencia del valor del *OID* (que indica el número de bits salientes) en un tiempo determinado y el valor del mismo *OID* transcurridos 5 minutos. *Pandora FMS* permite definir este tipo de chequeo definiendo el tipo de dato del módulo "*incremental data*", y aplicará los cálculos correspondientes para representar el número por segundo. Dado que el valor viene en bytes, deberemos aplicar un factor de postproceso de 8 para transformarlo a bits.

- **Enlace de Radio (Subida):** *Throughput* (bits/s) de subida en el enlace de radio.
- **Pérdida de Paquetes Radio (Subida) y Pérdida de Paquetes Radio (Bajada):** Miden la pérdida de paquetes (%) del enlace de radio en subida y en bajada, respectivamente. Son módulos creados en la sección "[4.2.4.1.- Módulos MACH1000](#)" y adaptados a las antenas *BreezeNET* a posteriori. Para más detalle, conviene consultar dicha sección.
- **Enlace Ethernet (Bajada):** *Throughput* (bits/s) de bajada en el enlace *GE*. *OID* .1.3.6.1.2.1.2.2.1.16.2.
- **Enlace Ethernet (Subida):** *Throughput* (bits/s) de subida en el enlace *GE*. *OID* .1.3.6.1.2.1.2.1.1.16.2.

- **Cifrado de Enlace de Radio:** Devuelve 2 si se encuentra activado y 1 si se encuentra desactivado. Por tanto, mostrará *critical* si devuelve un valor distinto de 2. *OID* .1.3.6.4.1.12394.1.1.9.3.0.
- **Autenticación con Clave:** Del mismo modo, esta consulta al *OID* .1.3.6.4.1.12394.1.1.9.3.0 muestra si el proceso de autenticación requiere una clave (2) o si es una red abierta (1). Indicaremos *critical* si la red se encuentra abierta.
- **Algoritmo de Cifrado:** Pudiendo ser 1 para *WEP* o 2 para. Consideraremos *warning* el uso de *WEP*. El *OID* es .1.3.6.4.1.12394.1.1.9.5.0.
- **Nivel de Ruido:** Ruido de la suma de todas aquellas señales ajenas al enlace de radio, medido en dB. Su estado siempre será correcto. *OID* .1.3.6.1.4.1.12394.1.1.11.11.0.
- **Ganancia de Antena:** Ganancia de la antena de radio, medida en *dBi*. *OID* .1.3.6.1.4.1.12394.1.1.6.17.0.
- **SNR:** *SNR (Signal to Noise Ratio)* medio de las tramas recibidas por el equipo. Siguiendo las recomendaciones de la empresa (basadas en su conocimiento), un ratio de entre 9-16 *dB* supone un estado *warning* y 0-8 *dB* un estado *critical*. *OID* .1.3.6.1.4.1.12394.1.1.11.1.0. ***NOTA:** Solo aplicable a estaciones *RB*.

Asimismo, el fabricante relaciona los distintos valores del ratio señal/ruido con la iluminación de los 5 *LEDs* indicadores del propio aparato:

- 1 *LED*: 5-8 *SNR*
 - 2 *LED*: 8-15 *SNR*
 - 3 *LED*: 15-20 *SNR*
 - 4 *LED*: 20-25 *SNR*
 - 5 *LED*: 25+ *SNR*
- **RSSI:** Indicador de la potencia de la señal recibida (*Received Signal Strength Indicator*), medido en *dBm*. Siguiendo las recomendaciones de la empresa, consideramos correctos los valores mayores que -60 *dBm*, *warning* los que se encuentran entre -70 y -76 y *critical* para aquellos valores que sean menores que -77. *OID* .1.3.6.1.4.1.12394.1.1.6.11.4.1.22.1. ***NOTA:** Solo válido para *RB*.
 - **Equipos de Radio Asociados:** Muestra el número de equipos de radio asociados a la unidad. Dado que se trata de una conexión punto a punto, un valor distinto de 1 correspondería a una situación anómala. *OID* .1.3.6.1.4.1.12394.1.1.1.13.0. ***NOTA:** Esta consulta solo será válida para las estaciones *BU*.

- **Banda de Radio:** Pudiendo tomar valores entre el 1 y el 9 para representar la banda de radio utilizada:
 - (1) 5.8 GHz
 - (2) 5.4 GHz
 - (3) 4.9 GHz
 - (4) 5.2 GHz
 - (5) 2.4 GHz
 - (6) 5.3 GHz
 - (7) 4.9 GHz
 - (8) 0.9 GHz

4.1.4.2.- Módulos BreezeULTRA

La organización de la información accesible vía *SNMP* es radicalmente distinta a los modelos *BreezeNET* vistos anteriormente: carecen de ciertos *MIBs* comunes a la mayoría de dispositivos de red (*MIB-2*, *Bridge-MIB*, *IF-MIB*) y emplea ciertos *MIBs* no documentados públicamente que han sido obtenidos directamente desde el proveedor. Por ello, la búsqueda de valores y *OIDs* útiles ha requerido una búsqueda más exhaustiva con el software *iReasoning MIB Browser* mencionado anteriormente.

Los módulos creados son:

- **Host Alive:** el clásico *ping*, cada 2 minutos.
- **Enlace PoE In (Estado):** Estado del enlace *PoE In*. Devuelve *critical* si se encuentra en estado *down*. *OID* .1.3.6.1.4.1.12394.1.50.10.2.12.0.
- **Enlace PoE Out (Estado):** Estado del enlace *PoE Out*. Dado que en estas antenas no se encuentra activo (se utiliza la interfaz *PoE In* en modo *Full Duplex*), este módulo estará definido pero se encontrará desactivado. *OID* .1.3.6.1.4.1.12394.1.50.10.2.11.0.
- **Enlace PoE In (Bajada):** Uso del enlace *PoE In* de bajada, en bits/s. Se aplica un factor de post-proceso de 8. *OID* .1.3.6.1.4.1.12394.1.50.10.7.1.1.1.4.1.
- **Enlace PoE In (Subida):** Ídem. *OID* .1.3.6.1.4.1.12394.1.50.10.7.1.1.1.3.2.
- **Enlace PoE Out (Bajada):** Ídem. *OID* .1.3.6.1.4.1.12394.1.50.10.7.1.1.1.6.1.
- **Enlace PoE Out (Subida):** Ídem. *OID* .1.3.6.1.4.1.12394.1.50.10.7.1.1.1.3.1.
- **Pérdida de Paquetes PoE In (Bajada):** Indica la pérdida de paquetes en el enlace, expresada en paquetes/s. *OID*: .1.3.6.1.4.1.12394.1.50.10.7.1.1.1.7.2.

- **Pérdida de Paquetes PoE In (Subida):** Ídem. **OID:** .1.3.6.1.4.1.12394.1.50.10.7.1.1.1.4.2.
- **Pérdida de Paquetes PoE Out (Bajada):** Ídem. **OID:** .1.3.6.1.4.1.12394.1.50.10.7.1.1.1.7.1.
- **Pérdida de Paquetes PoE Out (Subida):** Ídem. **OID:** .1.3.6.1.4.1.12394.1.50.10.7.1.1.1.4.1.
- **Potencia de Transmisión Sector 1:** Potencia de transmisión de la antena interna⁵ (en dBm). Si la antena no está instalada, devuelve valor 0. **OID** .1.3.6.1.4.1.12394.1.50.10.5.5.1.7.1.
- **Enlace Radio Sector 1 (Bajada):** Uso del enlace de radio de la antena interna de bajada, en bits/s (con multiplicador de 8, para transformar bytes en bits). **OID:** .1.3.6.1.4.1.12394.1.50.10.7.1.3.1.5.3.2.
- **Enlace Radio Sector 1 (Subida):** Ídem. **OID:** .1.3.6.1.4.1.12394.1.50.10.7.1.3.1.5.2.3.
- **Errores CRC Sector 1 (Bajada):** Número de paquetes con errores **CRC** que se reciben en la antena interna, por segundo. **OID** .1.3.6.1.4.1.12394.1.50.10.7.1.2.1.4.1.
- **Paquetes Retransmitidos Sector 1 (Subida):** Número de paquetes por segundo que han sido retransmitidos a través del enlace de radio del sector 1. **OID** .1.3.6.1.4.1.12394.1.50.10.7.1.2.1.9.1.
- **RSSI:** Indicador de la potencia de la señal recibida (*Received Signal Strength Indicator*), medido en *dBm*. Siguiendo las recomendaciones de la empresa, consideramos correctos los valores mayores que -60 *dBm*, *warning* los que se encuentran entre -70 y -76 y *critical* para aquellos valores que sean menores que -77. **OID** .1.3.6.1.4.1.12394.1.50.10.7.2.1.1.7.0.3.64.157.11.102. ***NOTA:** Solo válido para *BU*.
- **SNR de Señal Recibida:** Ratio señal/ruido de la señal recibida. **OID:** 1.3.6.1.4.1.12394.1.50.10.7.2.1.1.6.1.0.3.64.157.11.102. De nuevo, un ratio de entre 9-16 *dB* supone un estado *warning* y 0-8 *dB* un estado *critical*. ***NOTA:** Solo válido para *BU*.
- **Cifrado de Enlace de Radio (Sector 1):** Muestra valor 0 si se encuentra desactivado o valor 2 si la conexión está cifrada mediante *AES128*. Se mostrará *critical* siempre que la conexión no se encuentre cifrada. **OID:** .1.3.6.1.4.1.12394.1.50.10.5.11.1.2.1.

⁵ Los modelos *BreezeULTRA*, al soportar la configuración punto-multipunto, pueden contar con hasta 3 sectores. Aunque solo generemos módulos relativos al primer sector (el utilizado en punto a punto), la extrapolación a otros sectores es una tarea trivial.

- **Servidor Web:** Consulta *HTTP* al puerto 80, con lo que se verifica que el puerto está activo y que el servidor responde adecuadamente a la petición *HTTP GET /login.html*. Se mostrará *critical* cuando no sea así o cuando el código de retorno *HTTP* no sea 200 (OK).

El estado final de los agentes queda reflejado en la siguiente captura de pantalla, donde se muestra que todos los módulos están siendo monitorizados y se encuentran en estado correcto, salvo el módulo *RSSI* de la antena *BreezeULTRA_BU1*:

Agent	Description	OS	Interval	Group	Modules	Status	Alerts
BreezeNET_BU1	Antena BreezeNET Base Unit 1		5 minutes		14 : 14		
BreezeNET_RB1	Antena BreezeNET Remote Bridge 1		5 minutes		15 : 15		
BreezeULTRA_BU1	Antena BreezeULTRA Base Unit 1		5 minutes		15 : 1 : 14		
BreezeULTRA_RB1	Antena BreezeULTRA Remote Bridge 1		5 minutes		13 : 13		

4.1.5.- Configuración de Alertas

No existe especial interés en generar un tipo de alerta determinada para estos dispositivos, por lo que nos limitaremos a definir una alerta que envíe un correo electrónico cuando el módulo *Host Alive* del agente *BreezeULTRA_BU1* se encuentre en estado *critical* tras dos chequeos consecutivos.

Agent	Module	Template	Actions	Status
BreezeULTRA_BU1	Host Alive	Critical condition	Mail to ander.[...]s@gmail.com (On 1) Add action	

4.1.6.- Evaluación de Soluciones

Puesto que las antenas se encuentran operativas y forman parte de la infraestructura de red de un cliente de la empresa, encontramos una limitación del tipo de pruebas que se podrían realizar, descartando aquellas que puedan afectar a la continuidad del funcionamiento de estas antenas (parada de servicios, reducción de la potencia de señal, y demás simulaciones "agresivas" de otras condiciones).

Decidimos llevar a cabo las siguientes pruebas: forzar la imposibilidad de acceso del *Pandora FMS Server* a las antenas (cerrando el túnel *VPN* y analizando la respuesta de la aplicación) y contrastar la información obtenida por *Pandora* con la ofrecida por el servicio web de las antenas *BreezeULTRA*.

4.1.6.1.- Contraste de Información con el Servidor Web BreezeULTRA

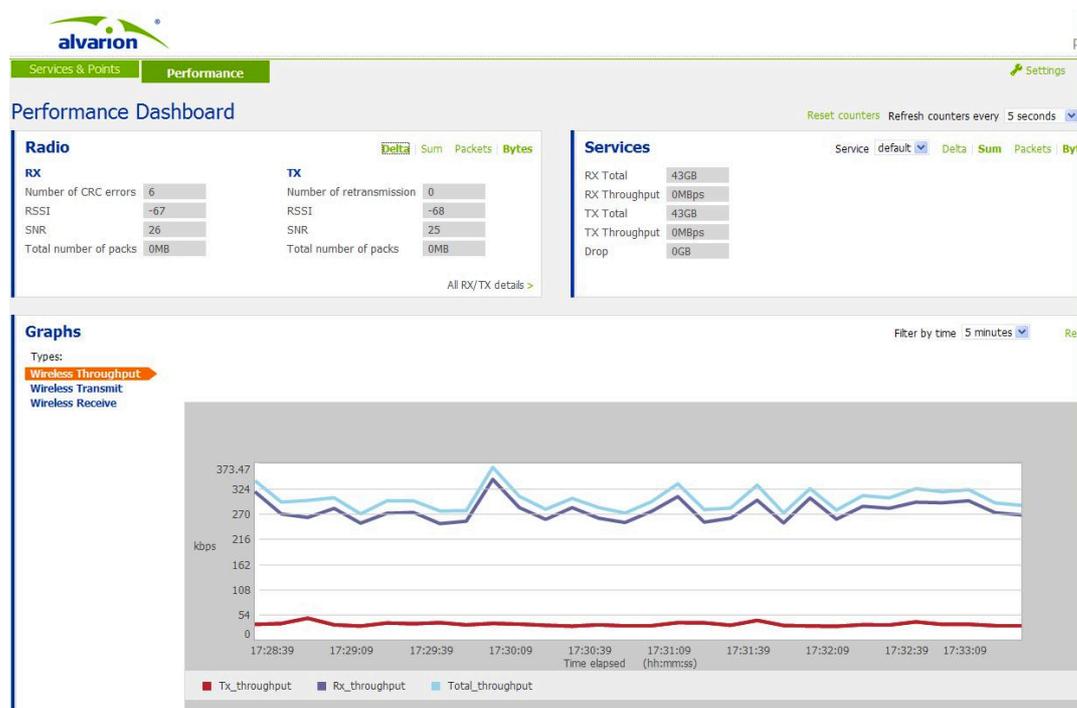
Tal y como se ha mencionado en apartados anteriores, el fabricante *Alvarion* incluyó en sus modelos *BreezeULTRA* una interfaz web para monitorizar y gestionar de manera individual de cada antena.

Se ha contrastado la información de todos los módulos creados con la información presentada a través de la interfaz web (salvo los relativos al

cifrado de la conexión, ya que no están presentes), corrigiendo aquellos módulos no concordantes.

La interfaz web ha servido, además, para obtener ideas sobre nuevos valores interesantes y susceptibles de ser monitorizados, dando como resultado final la lista de módulos para antenas *BreezeULTRA*.

El aspecto de dicha interfaz es el siguiente:



4.1.6.2.- Cierre del Túnel VPN

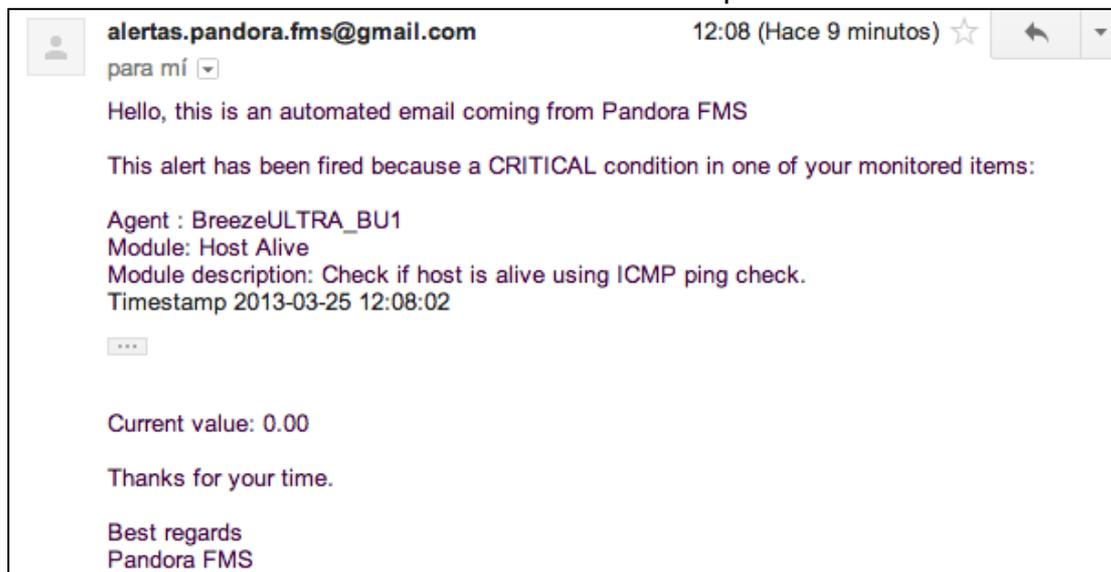
En este apartado, forzaremos una situación de error del sistema para evaluar la respuesta de *Pandora FMS*: el estado de los módulos, estado de los agentes lógicos, ejecución de alertas, etcétera.

Para ello, cerramos el túnel *VPN* que interconecta el servidor *Pandora FMS* y la red en la que se encuentran las antenas *Alvarion*. Tras unos minutos desde el cierre del túnel, el aspecto de la consola web de *Pandora* es el siguiente:

Agent	Description	OS	Interval	Group	Modules	Status	Alerts
BreezeNET_BU1	Antena BreezeNET Base Unit 1		5 minutes		14 : 1 : 11 : 2		
BreezeNET_RB1	Antena BreezeNET Remote Bridge 1		5 minutes		15 : 1 : 12 : 2		
BreezeULTRA_BU1	Antena BreezeULTRA Base Unit 1		5 minutes		15 : 1 : 2 : 13		
BreezeULTRA_RB1	Antena BreezeULTRA Remote Bridge 1		5 minutes		13 : 2 : 11		

Observamos que todos los agentes se encuentran en estado *critical* ya que, al menos, uno de sus módulos se encuentra en dicho estado (en este caso, *Host Alive* y *Servidor Web*). Las consultas *SNMP* fallidas muestran un estado *unknown* o desconocido (en color gris).

Asimismo, se muestra que la alerta que definimos anteriormente para el módulo *Host Alive* en el agente *BreezeULTRA_BU1* ha sido disparada. Este es el correo electrónico enviado automáticamente por *Pandora*:



Se ha mantenido el cuerpo de la plantilla estándar de correo electrónico, si bien es posible modificarla y personalizarla fácilmente desde la consola web de *Pandora FMS*, mediante una combinación de texto y macros que representen ciertos valores determinados de la alerta.

Consideramos que la personalización de los correos no resulta de interés para el desarrollo del proyecto, por lo que mantendremos las plantillas estándar.

4.2.- Equipamiento Networking Industrial Hirschmann

4.2.1.- Análisis de Dispositivos

Hirschmann™ es una marca adquirida por *Belden Inc.* especializada en dispositivos de red como *switches*, *routers*, *hubs*, *APs WiFi* y etcétera para entornos industriales. Estos productos se diferencian de los de otras marcas como *Cisco* en su robustez y capacidad para soportar condiciones ambientales desfavorables de temperatura, humedad, presión, corrosión y demás.

Los dispositivos concretos que la empresa desea monitorizar son los siguientes:

4.2.1.1.- MACH102-8TP-R

El *MACH102-8TP-R* es un *switch* industrial en formato *rack* que opera hasta el nivel 2 (de enlace) del modelo de capas *OSI*. Cuenta con 8 puertos *Fast Ethernet (FE, 100 Mbit/s)* y 2 *Gigabit Ethernet (GE, 1 Gbit/s)*, aunque pueden acoplarse módulos para llegar a los 24 puertos *FE*, tal y como se muestra en la siguiente imagen:



Cuenta con una fuente de alimentación redundante de 2 conectores, con lo que, conectados a distintos tramos de la red eléctrica o a distintas *UPS (Uninterruptible Power Supply)*, se consigue que el equipo tenga una mayor resistencia a apagones o desconexiones.

Soporta todas las versiones de *SNMP (v1, v2 y v3)*. Además de los *MIBs* tradicionales (*MIB-2, Bridge-MIB*), el fabricante ofrece hasta 20 ficheros *MIB* privados que serán analizados para desarrollar los módulos que consideremos útiles.

Por tanto, mediante chequeos externos, nos interesaría monitorizar:

- Conectividad del dispositivo (*Ping*).
- Estado de las interfaces en uso (activas, caídas, ...).
- Tráfico (bit/s) entrante y saliente de cada interfaz utilizada.
- Pérdida de paquetes de cada interfaz utilizada.
- Estado de las fuentes de alimentación (conectadas, caídas, ...).
- Otras condiciones ambientales.

4.2.1.2.- MACH1000 Full GE L3P PoE

Este dispositivo, de gama más alta y ostensiblemente más caro que el *switch* anterior, cuenta con 16 puertos *GE*, siendo 4 de ellos puertos *PoE* opcionales

para alimentarlo eléctricamente además de las 2 fuentes de alimentación. Puede operar hasta el nivel 3 (de red) del modelo de capas OSI.



Tal y como sucede en el modelo anterior, disponemos de una gran cantidad de *MIBs* propietarios además de los genéricos, pudiendo consultarse mediante *SNMPv1*, *v2* y *v3*.

Análogamente, mediante chequeos externos, nos interesaría monitorizar:

- Conectividad del dispositivo (*Ping*).
- Estado de las interfaces en uso (activas, caídas, ...).
- Tráfico (bit/s) entrante y saliente de cada interfaz utilizada.
- Pérdida de paquetes de cada interfaz utilizada.
- Estado de las fuentes de alimentación y de los puertos *PoE* (conectados, caídos, ...).
- Otras condiciones ambientales.

4.2.2.- Dispositivos de Prueba

Se nos otorga acceso (mediante el mismo túnel *VPN*), a los siguientes dispositivos que denominamos:

Equipos **MACH1000**:

- *MACH1000_1*
- *MACH1000_2*
- *MACH1000_3*
- *MACH1000_4*
- *MACH1000_5*
- *MACH1000_6*

Equipos **MACH100**:

- *MACH102_1*
- *MACH102_2*

4.2.3.- Creación de Agentes

Desde la consola web de *Pandora FMS*, creamos los agentes lógicos correspondientes a los equipos *MACH102_1* y *MACH1000_1*.

4.2.4.- Desarrollo de Módulos

Generaremos manualmente los módulos asociados a los agentes recién creados. Aunque muchos de ellos serán idénticos para ambos modelos, optamos por dividir esta sección en módulos para *MACH102* y módulos para *MACH1000*.

De nuevo, empleamos el *iReasoning MIB Browser* para cargar los *MIBs* propietarios del fabricante *Hirschmann* y navegar a través de ellos con comodidad, para posteriormente obtener los *OIDs* deseados.

Al tratarse de *switches* con un gran número de puertos, es previsible que se generen multitud de módulos asociados a puertos que no estén siendo utilizados. Optamos por generarlos igualmente y, después, procederemos a desactivar aquellos módulos correspondientes a interfaces físicas no utilizadas (tarea simple mediante "*Massive operations*", funcionalidad de *Pandora FMS*).

4.2.4.1.- Módulos MACH1000

Los módulos generados para los *switches Hirschmann MACH102* son los siguientes:

- **Host Alive:** *Ping* que comprueba que el dispositivo sigue activo. Se chequea cada 2 minutos.
- **CPU (30 min):** Uso de CPU (%) medio registrado durante 30 minutos. Lo consideraremos *warning* cuando supere el 80% y *critical* cuando supere el 90% . *OID* .1.3.6.1.4.1.248.14.2.15.2.2.0.
- **Temperatura:** Valor que entrega el sensor térmico del *switch*, en °C. Dado que el fabricante indica una temperatura de operación de entre 0 y 50 °C, consideraremos *warning* cuando esta supere los 40°C y *critical* cuando supere los 50°C. *OID* .1.3.6.1.4.1.248.14.2.5.1.0.
- **Fuente de Alimentación (1):** Estado de la fuente de alimentación número 1. Puede ser:
 - (1) *OK*
 - (2) *Failed*
 - (3) *NotInstalled*
 - (4) *Unknown*

Mostraremos *critical* cuando su estado sea *Failed*, *NotInstalled* o *Unknown*. *OID* .1.3.6.1.4.1.248.14.1.2.1.3.1.1.

- **Fuente de Alimentación (2).** Ídem. *OID* .1.3.6.1.4.1.248.14.1.2.1.3.1.2.
- **Potencia Máxima PoE:** Potencia máxima (en *wattios*) que el *switch* puede suministrar a través de sus puertos *PoE* (en suma). Su estado es siempre *OK*. *OID* .1.3.6.1.4.1.248.14.2.14.1.3.0.

- **Alimentación PoE (1):** Potencia (en *wattios*) que suministra el puerto PoE número 1 del *switch*. Nunca se considera una situación de error. *OID* .1.3.6.1.4.1.248.14.2.14.2.1.2.1.2.1.
- **Alimentación PoE (2):** Ídem. *OID* .1.3.6.1.4.1.248.14.2.14.2.1.2.1.2.2.
- **Alimentación PoE (3):** Ídem. *OID* .1.3.6.1.4.1.248.14.2.14.2.1.2.1.2.3.
- **Alimentación PoE (4):** Ídem. *OID* .1.3.6.1.4.1.248.14.2.14.2.1.2.1.2.4.
- **Estado Puerto GE 1:** Muestra *critical* cuando el estado del puerto no es *OK*. *OID* .1.3.6.1.2.1.2.2.1.8.1.
- **Estado Puerto GE N:** Ídem para cualquier otro índice N del puerto (hasta 16). *OID* .1.3.6.1.2.1.2.2.1.8.N.
- **Enlace GE 1 (Bajada):** Bits por segundo de bajada en la interfaz GE 1. Se aplica un valor de postproceso de 8 (para transformar bytes en bits) y se chequea cada 3 minutos. Siempre se encuentra en estado *OK*. *OID* .1.3.6.1.2.1.2.2.1.10.1.
- **Enlace GE N (Bajada):** Ídem para cualquier otro índice N del puerto (hasta 16). *OID* .1.3.6.1.2.1.2.2.1.10.N.
- **Enlace GE 1 (Subida):** Bits por segundo de Subida en la interfaz GE 1. Se aplica un valor de postproceso de 8 (para transformar bytes en bits) y se chequea cada 3 minutos. Siempre se encuentra en estado *OK*. *OID* .1.3.6.1.2.1.2.2.1.16.1.
- **Enlace GE N (Subida):** Ídem para cualquier otro índice N del puerto (hasta 16). *OID* .1.3.6.1.2.1.2.2.1.16.N.
- **Pérdida de Paquetes Puerto GE 1 (Bajada):** Mide la pérdida de paquetes (%) de bajada de la interfaz GE 1. El desarrollo de este módulo ha exigido la integración de un *plug-in* del repositorio oficial de *Pandora FMS*. El proceso de integración ha sido el siguiente:

- Descargamos el *script* "*perdidos.pl*" (en este caso, en lenguaje *Perl*) y lo ubicamos en la siguiente ruta del *Pandora FMS Server*:

```
/usr/share/pandora_server/util/plugin/perdidos_b.pl
```

Le concedemos permisos de ejecución globales, mediante el comando *chmod*. La documentación indica que el *script* puede ejecutarse de la siguiente manera:

```
perdidos.pl -h <@ip> -c <comunidad_v2> -p <interfaz>
```

Analizando el código, observamos que agrupa los valores de varias *OIDs* de la tabla de interfaces *IF-TABLE* (paquetes descartados, erróneos, desconocidos...) de bajada.

Name	Pérdida de Paquetes (bajada)
Plug-in command	/usr/share/pandora_server/util/plugin/perdidos_t
Plug-in type	Standard
Max. timeout	0
IP address option	-h
Port option	-p
User option	
Password option	-c
Description	Pérdida de paquetes de una interfaz (%) en bajada. Agrupa varias <i>OIDs</i> del <i>IF-MIB</i> (estándar). Puerto == Interfaz Password == Comunidad

- A través de la consola web, accedemos al servidor de *plug-ins* y creamos "*Pérdida de Paquetes (bajada)*":
- Y, por último, creamos el módulo (*Plug-In Server Module*) asociado al agente lógico correspondiente al *switch*. Para mostrar un valor en %, marcamos un ratio de postproceso de 100.
- **Pérdida de Paquetes Puerto GE N (Bajada):** Ídem para cualquier interfaz física de índice N. Basta con reutilizar el *plug-in* creado anteriormente.
- **Pérdida de Paquetes Puerto GE 1 (Subida):** Análogamente, mide la pérdida de paquetes (%) de subida de la interfaz *GE 1*. De nuevo, necesitaremos crear otro *plug-in* personalizado e integrarlo. Se han seguido los siguientes pasos:
 - Modificamos el original "*perdidos.pl*" para obtener "*perdidos_s.pl*". A grandes rasgos, los cambios introducidos son los siguientes:
 - Sustitución de *OIDs* que representan valores de entrada por *OIDs* análogas que representan valores de salida.
 - A raíz del punto anterior, dejamos de considerar la pérdida de paquetes causada por protocolos desconocidos. El *IF-MIB* no representa este valor, ya que no tiene sentido para paquetes salientes⁶.

⁶ Profundizando en este asunto, se debe a que cuando el *switch* detecte un protocolo incorrecto o desconocido, no retransmitirá el paquete e imputará la pérdida a la interfaz por la que haya entrado.

- Cambio de nombres de los ficheros temporales empleados, a fin de que no colisionen con el *script* original.
 - Insertamos el *script* en el *Pandora FMS Server* en la ruta descrita y le otorgamos permiso de ejecución:


```
/usr/share/pandora_server/util/plugin/perdidos_s.pl
```
 - Del mismo modo, creamos el *plug-in* "*Pérdida de Paquetes (subida)*" dentro del *Plug-In Server* desde la consola web de *Pandora FMS*.
 - Por último, creamos el *Plug-In Server Module* asociado al agente lógico, marcando un ratio de postproceso de 100.
- **Pérdida de Paquetes Puerto GE N (Subida):** Ídem para cualquier interfaz física de índice N. Reutilizamos el *plug-in* "*Pérdida de Paquetes (subida)*" creado anteriormente.

Observamos los ficheros temporales creados por los *plug-ins* (distinguidos por la dirección *IP* del agente lógico, seguido del índice de la interfaz monitorizada y de *paq/paq2* para diferenciar la pérdida de paquetes en bajada y en subida):

```

root@pandoraserver ~]# ls /tmp
192.168.1.208:10:paq  192.168.1.208:6:paq2
192.168.1.208:11:paq2 192.168.1.208:7:paq
192.168.1.208:12:paq  192.168.1.208:7:paq2
192.168.1.208:12:paq2 192.168.1.208:8:paq
192.168.1.208:13:paq  192.168.1.208:9:paq
192.168.1.208:13:paq2 192.168.1.208:9:paq2
192.168.1.208:14:paq  netExtender.log
192.168.1.208:15:paq2 netextender.pid
192.168.1.208:16:paq  pandora_serialize_admin_snmp__1__1
192.168.1.208:1:paq   pppenv
192.168.1.208:1:paq2  vmware-config8
192.168.1.208:2:paq   vmware-config1
192.168.1.208:2:paq2  VMwareDmD
192.168.1.208:3:paq   vmware-fonts8
192.168.1.208:3:paq2  vmware-fonts1
192.168.1.208:4:paq   vmware-root
192.168.1.208:4:paq2  vmware-root-4241867844
192.168.1.208:5:paq   yum.log
192.168.1.208:5:paq2  yum_save_tx-2013-03-25-10-46S1eB_2.yumtx
192.168.1.208:6:paq

```

Durante el ciclo de vida de la aplicación, es muy probable que se creen y se destruyan módulos asociados a la pérdida de paquetes, con la consiguiente acumulación de ficheros temporales inservibles en el directorio */tmp*.

Considerando este hecho, programamos una tarea de mantenimiento periódica que se encargará de eliminar los ficheros generados por los *plug-ins* mencionados que no hayan sido utilizados recientemente, favoreciendo así la limpieza y estabilidad del sistema.

Mediante el comando "*crontab -e*", escribimos:

```
0 12 * * * find /tmp -name "*. *.*.*:*:paq*" -ctime +1 -exec rm
    {} \;
```

De este modo, cada día a las 12:00, se lanzará esta tarea que buscará y eliminará todos aquellos ficheros que se encuentren bajo el directorio */tmp*, cuyo nombre posea el formato definido en los *plug-ins* y que no haya sido modificado en, al menos, un día.

- **Uso de Potencia *PoE*:** Muestra la relación (en %) entre la suma de los *wattios* suministrados por el *switch* y los *wattios* máximos que puede suministrar. Muestra *warning* al superar el 75% de uso, indicando que el *switch* apenas puede suministrar más potencia.

Este módulo emplea un *plug-in* desarrollado para este cometido específico, basado en el *script* "*hirschmann_uso_poe.sh*", de modo similar al caso anterior.

Este *script* realiza consultas *SNMP* para calcular la relación entre la suma de las potencias de los puertos *PoE* y la potencia total que el *switch* es capaz de suministrar.

- **Cambio Topológico *STP*:** Muestra hace cuántas horas se realizó el último cambio sobre la topología de la red a causa del *Spanning Tree Protocol*. Una reconfiguración demasiado frecuente puede ser tanto causa de una bajada de rendimiento de la red como síntoma de un mal funcionamiento de otro sistema, que fuerza la ejecución del *STP*.

Para indicar un cambio reciente, mostraremos *warning* cuando este tiempo sea inferior a 1h. El chequeo se realizará cada 30 minutos, indicando un valor de postproceso de $2,777778 \cdot 10^{-6}$ para transformar "*timeticks*" del *switch* (centésimas de segundo) en horas. *OID* .1.3.6.1.4.1.248.15.1.2.15.8.12.0.

4.2.4.2.- Módulos *MACH102*

Para el *MACH102*, generamos los siguientes módulos:

- ***Host Alive*:** Idéntico al caso anterior.
- ***CPU (30 min)*:** Idéntico al caso anterior
- ***Temperatura*:** Idéntico al caso anterior
- ***Fuente de Alimentación (1)*:** Idéntico al caso anterior
- ***Fuente de Alimentación (2)*:** Idéntico al caso anterior

- **Estado Módulo 1 Puerto GE N:** Estado de los puertos *Gigabit Ethernet* del *switch*. Solo se incluyen los puertos 1 y 2, ya que son los únicos que soportan *Gigabit Ethernet*.
- **Estado Módulo 2 Puerto FE N:** Estado de los puertos *Fast Ethernet (100BaseT)* del *switch* del módulo #2, para un índice N entre 1 y 8 incluidos.
- **Estado Módulo 3 Puerto FE N:** Ídem para el módulo #3.
- **Enlace Módulo 1 Puerto GE N (Bajada):** Tráfico de bajada que atraviesa el puerto 1 o 2 GE del módulo #1, en *bits/s*.
- **Enlace Módulo 2 Puerto FE N (Bajada):** Ídem para los puertos FE del módulo #2.
- **Enlace Módulo 3 Puerto FE N (Bajada):** Ídem para el módulo #3.
- **Enlace Módulo 1 Puerto GE N (Subida):** Tráfico de subida que atraviesa el puerto 1 o 2 GE del módulo #1, en *bits/s*.
- **Enlace Módulo 2 Puerto FE N (Subida):** Ídem para los puertos FE del módulo #2.
- **Enlace Módulo 3 Puerto FE N (Subida):** Ídem para el módulo #3.
- **Pérdida de Paquetes (Bajada) Módulo 1 Puerto GE N:** Pérdida de paquetes (en paquetes/s) de bajada en el puerto de índice N del módulo 1.
- **Pérdida de Paquetes (Bajada) Módulo 2 Puerto FE N:** Ídem para los puertos FE del módulo #2.
- **Pérdida de Paquetes (Bajada) Módulo 3 Puerto FE N:** Ídem para el módulo #3.
- **Pérdida de Paquetes (Subida) Módulo 1 Puerto GE N:** Pérdida de paquetes (en paquetes/s) de subida en el puerto de índice N del módulo 1.
- **Pérdida de Paquetes (Subida) Módulo 2 Puerto FE N:** Ídem para los puertos FE del módulo #2.
- **Pérdida de Paquetes (Subida) Módulo 3 Puerto FE N:** Ídem para el módulo #3.
- **Cambio Topológico STP:** Idéntico al *switch* anterior.

4.2.5.- Configuración de Alertas

Por similitud con las antenas *WiMAX* (y existiendo ya una alerta-tipo de notificación por *email*), no se considera interesante configurar ningún nuevo tipo de alerta para este equipamiento.

4.2.6.- Evaluación de Soluciones

Nos encontramos con la misma limitación de las antenas: las pruebas a realizar no pueden afectar al funcionamiento normal de los *switches*, ya que forman parte de una red activa.

4.2.6.1.- Limpieza de Ficheros No Utilizados

Realizamos una prueba simple, que consistirá en crear dos ficheros (antes de la hora programada de la tarea, las 12:00):

```
# touch /tmp/111.111.111.111:1:paq
# touch /tmp/111.111.111.111:1:paq2
```

Y esperar al siguiente día. La tarea ya se habrá ejecutado una vez, aunque, dado que la fecha de modificación indica que los ficheros han sido modificados (creados, en realidad) en menos de un día, los ficheros no habrán sido eliminados.

Antes de las 12:00, actualizamos la fecha de modificación de uno de los dos ficheros:

```
# touch /tmp/111.111.111.111:1:paq2
```

```
[root@pandoraserver ~]# ls -l /tmp | grep 111
-rw-r--r-- 1 root root 0 Apr 4 09:24 111.111.111.111:1:paq
-rw-r--r-- 1 root root 0 Apr 5 08:58 111.111.111.111:1:paq2
```

Pasada la hora de ejecución del "*cron job*", comprobamos que el fichero modificado por última vez hace más de un día ha sido eliminado.

4.2.6.2.- Pérdida de Paquetes (Plug-In)

Dado que la pérdida de paquetes en los *switches* es mínima y que provocar la pérdida de paquetes a través de una interfaz no resulta factible, simplemente comprobaremos que las cifras mostradas por estos *plug-ins* se acercan a la realidad.

Para ello, nos centramos en la interfaz del *MACH1000_1* que, por alguna razón, muestra más pérdida de paquetes (errores de bajada, indicados en el *OID IfInErrors*) en el software *MIB Browser*, que en este caso es *GE2*:

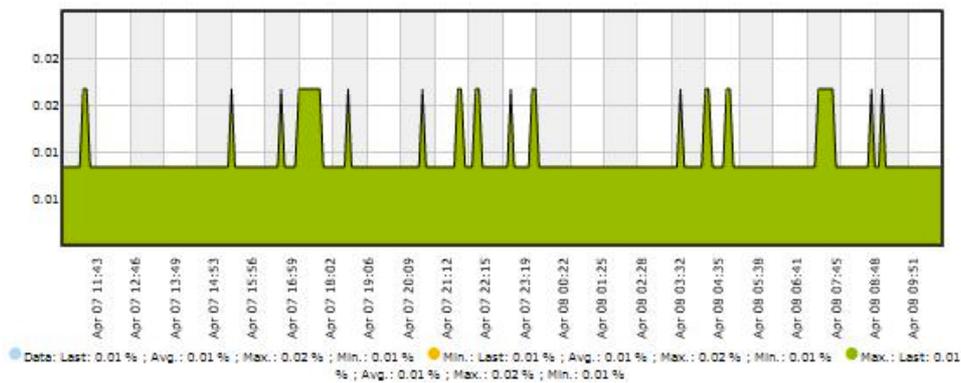
	1	2	3	4	5
ifIndex	1	2	3	4	5
ifDescr	Module: 1 Port: 1...	Module: 1 Port: 2...	Module: 1 Port: 3...	Module: 1 Port: 4...	Module: 1 F
ifType	6	6	6	6	6
ifMtu	1500	1500	1500	1500	1500
ifSpeed	100000000	100000000	0	0	0
ifPhysAddress	00-80-63-F5-4C-08	00-80-63-F5-4C-09	00-80-63-F5-4C-0A	00-80-63-F5-4C-0B	00-80-63-F
ifAdminStatus	up	up	up	up	up
ifOperStatus	up	up	down	down	down
ifLastChange	5585 hours 28 mi...	5083 hours 48 mi...	3876 hours 47 mi...	0 millisecond	0 millisecond
ifInOctets	681143624	267193887	59016	0	0
ifInUcastPkts	5661184	2092989747	0	0	0
ifInNUcastPkts	825471	91299	331	0	0
ifInDiscards	0	0	0	0	0
ifInErrors	0	652554	0	0	0
ifInUnknownProtos	0	0	0	0	0
ifOutOctets	3179318105	1070285584	83462	0	0
ifOutUcastPkts	12576097	3397015051	85	0	0
ifOutNUcastPkts	490123523	186718458	647	0	0
ifOutDiscards	0	0	0	0	0
ifOutErrors	0	0	0	0	0
ifOutQLen	0	0	0	0	0
ifSpecific	.0.0	.0.0	.0.0	.0.0	.0.0

Comparamos los datos con otro de esta tabla (el tiempo transcurrido entre los *snapshots* resulta indiferente para el cálculo, aunque en este caso es de aproximadamente 10 minutos):

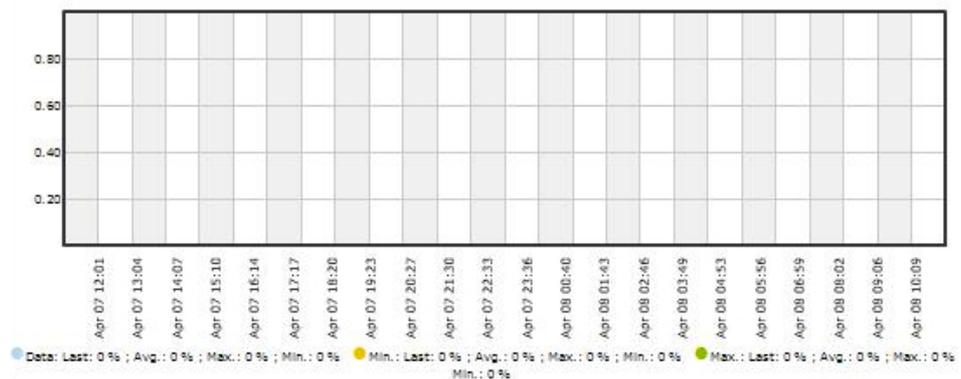
	1	2	3	4	5
ifIndex	1	2	3	4	5
ifDescr	Module: 1 Port: 1...	Module: 1 Port: 2...	Module: 1 Port: 3...	Module: 1 Port: 4...	Module: 1 F
ifType	6	6	6	6	6
ifMtu	1500	1500	1500	1500	1500
ifSpeed	100000000	100000000	0	0	0
ifPhysAddress	00-80-63-F5-4C-08	00-80-63-F5-4C-09	00-80-63-F5-4C-0A	00-80-63-F5-4C-0B	00-80-63-F
ifAdminStatus	up	up	up	up	up
ifOperStatus	up	up	down	down	down
ifLastChange	5585 hours 28 mi...	5083 hours 48 mi...	3876 hours 47 mi...	0 millisecond	0 millisecond
ifInOctets	681218057	1451603248	59016	0	0
ifInUcastPkts	5661800	2093808679	0	0	0
ifInNUcastPkts	825564	91310	331	0	0
ifInDiscards	0	0	0	0	0
ifInErrors	0	652622	0	0	0
ifInUnknownProtos	0	0	0	0	0
ifOutOctets	3182541957	1102280665	83462	0	0
ifOutUcastPkts	12577003	3397448253	85	0	0
ifOutNUcastPkts	490151666	186747319	647	0	0
ifOutDiscards	0	0	0	0	0
ifOutErrors	0	0	0	0	0
ifOutQLen	0	0	0	0	0
ifSpecific	.0.0	.0.0	.0.0	.0.0	.0.0

Lo que resulta en un 0'0083% de pérdida de paquetes en este intervalo de tiempo de prueba.

Aunque el resultado no resulta de validez como medida precisa, sí que podemos comprobar que los órdenes de estas magnitudes se corresponden con lo mostrado en la gráfica asociada al módulo "*Pérdida de Paquetes GE2 (Bajada)*":



Que indica una pérdida de paquetes media de 0'01% con picos de algo más de 0'02%. Esta misma gráfica de pérdida de paquetes en el resto de interfaces (que es de 0% absoluto, ya que sus *OIDs* indican que nunca se ha perdido ningún paquete) se muestra vacía:



4.2.6.3.- Uso de Potencia PoE (Plug-In)

El correcto funcionamiento de este módulo resulta sencillo de comprobar, calculando la suma de las potencias que suministran los puertos *PoE* en relación con la potencia máxima que el *switch* puede suministrar.

Comprobamos el caso del *switch MACH1000_3*, donde solo un puerto *PoE* suministra 3'9W potencia a algún otro dispositivo y puede ofrecer hasta 16W en total:

MACH1000_3	SNMP DATA	Alimentación PoE (1)	5 minutes	100%		101	N/A - N/A	3.9 W
MACH1000_3	SNMP DATA	Alimentación PoE (2)	5 minutes	100%		101	N/A - N/A	0 W
MACH1000_3	SNMP DATA	Alimentación PoE (3)	5 minutes	100%		101	N/A - N/A	0 W
MACH1000_3	SNMP DATA	Alimentación PoE (4)	5 minutes	100%		101	N/A - N/A	0 W
MACH1000_3	SNMP DATA	Potencia Máxima PoE	5 minutes	100%		101	N/A - N/A	16 W
MACH1000_3	DATA	Uso de Potencia PoE	5 minutes	100%		101	0/75 - N/A	24 %

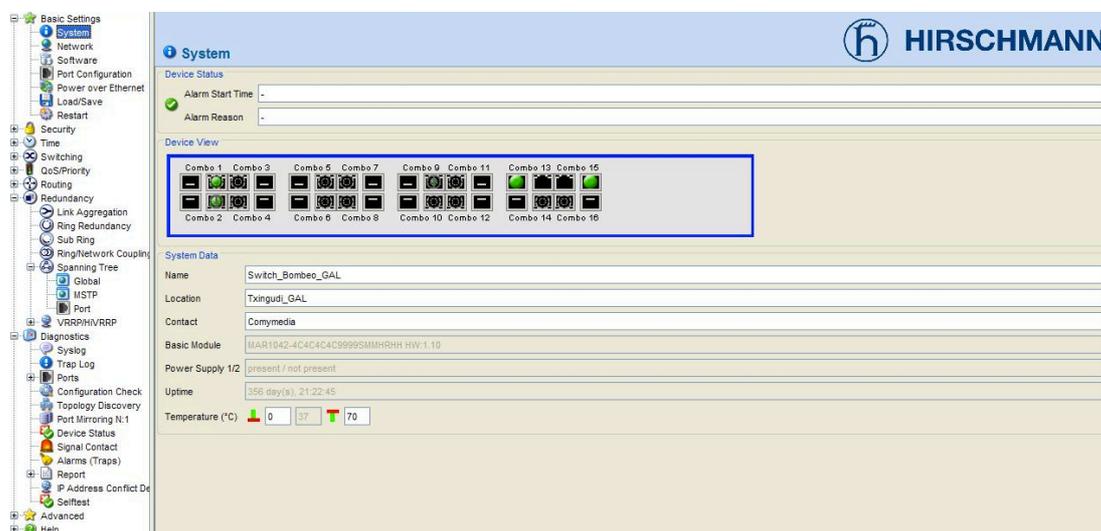
Y comprobamos un segundo caso en el que el *switch MACH1000_1* no reserva potencia a los puertos *PoE*⁷:

MACH1000_1	SNMP DATA	Alimentación PoE (1)	5 minutes			N/A - N/A	0 W
MACH1000_1	SNMP DATA	Alimentación PoE (2)	5 minutes			N/A - N/A	0 W
MACH1000_1	SNMP DATA	Alimentación PoE (3)	5 minutes			N/A - N/A	0 W
MACH1000_1	SNMP DATA	Alimentación PoE (4)	5 minutes			N/A - N/A	0 W
MACH1000_1	SNMP DATA	Potencia Máxima PoE	5 minutes			N/A - N/A	0 W
MACH1000_1	DATA	Uso de Potencia PoE	5 minutes			0/75 - N/A	100 %

4.2.6.4.- Contraste de Información con el Servidor Web

Se ha contrastado la información de los módulos relativos al uso de potencia y estado de puertos *PoE*, cambios provocados por el *STP*, temperatura e interfaces físicas activas.

La apariencia de la interfaz web de estos dispositivos es la siguiente:



4.2.6.5.- Fuentes de Alimentación

La empresa confirma que los *switches MACH1000* solo se alimentan a través de una fuente de alimentación (es decir, que no se utiliza la alimentación de respaldo), tal y como muestran los módulos:

Fuente de Alimentación (1)	(1) OK (2) Failed (3) NotInstalled (4) Unknown		N/A - 4/2	1
Fuente de Alimentación (2)	(1) OK (2) Failed (3) NotInstalled (4) Unknown		N/A - 4/2	3

⁷ Se ha decidido que en estas situaciones el módulo muestre un 100% de utilización, dando a entender que el *switch* no puede suministrar más potencia a través de puertos *PoE*.

4.3.- Servidores *HP ProLiant*

4.3.1.- Análisis de Dispositivos

Pasamos a analizar la línea de servidores de *Hewlett-Packard* orientados a entornos empresariales: los *HP ProLiant*.

Estos servidores están preparados para funcionar ininterrumpidamente, contando con fuentes de alimentación redundantes y múltiples bahías para discos duros, lo que permite establecer redundancia de datos mediante cualquiera de los niveles de *RAID*.

Asimismo, puede contar con 1 o 2 procesadores multinúcleo y varios *slots* de memoria *RAM* para soportar un único sistema operativo (*Windows*, *Linux*, *Solaris*, ...) o una infraestructura virtual (como *VMware ESX*), que albergue varias máquinas virtuales en el mismo servidor físico.

Estos servidores cuentan con un sistema de administración propio, muy potente, que resulta de gran interés para nuestro análisis: *HP iLO (Integrated Lights-Out) Management Engine*. Dedicaremos un apartado específico para analizar esta herramienta.

Pasamos analizar con más profundidad los modelos que le resultan interesantes a la empresa (ya que están operativos en uno de sus clientes):

4.3.1.1.- *HP ProLiant DL360*

Este es el clásico servidor empresarial en formato *rack*, para ser colocado junto con otros servidores o dispositivos de red en un *armario-rack* dentro de un *CPD* (Centro de Procesamiento de Datos).



Cuenta con 2 *sockets* para procesadores *Intel Xeon* multinúcleo y con 18 slots de memoria *RAM*. Como dato, su configuración más potente contaría con 2 procesadores de 6 núcleos y 384 *GB* de memoria *RAM*.

Posee 8 bahías para los nuevos *SSD* o para los tradicionales discos duros. Estos pueden extraerse y recolocarse cómodamente desde el panel frontal, lo que favorece el *hot swap* (reemplazo de discos en caliente, sin desconectar el servidor) en sistemas *RAID*.

En cuanto a la alimentación eléctrica, puede contar con un máximo de 2 fuentes de alimentación redundantes.

Una de las cinco tarjetas de red está reservada para acceder a *iLO3*, la herramienta de gestión maestra de este servidor. También cuenta con otros

elementos que no nos resultarán especialmente interesantes: slots *PCI-E*, puertos *USB*, salidas *VGA*, ...

4.3.1.2.- HP ProLiant DL380

Es el hermano mayor del *ProLiant DL360*, muy similar en aspectos técnicos. Podemos advertir una mejora de la conectividad del equipo (puertos *PCI-E*) y un mayor grosor (ocupando el equivalente a 2 *racks* tradicionales).

Los nuevos modelos G8 de octava generación pueden contar con 8 bahías adicionales (un total de 16) para dispositivos de almacenamiento:



4.3.1.3.- Plataforma HP Integrated Lights-Out (iLO)

El subsistema *iLO*, en sus distintas versiones, se compone de un microcontrolador inteligente con una memoria e interfaz de red dedicada. Esto hace que *iLO* funcione de manera totalmente independiente al servidor o al sistema operativo que se esté ejecutando sobre él.



Mediante las versiones más recientes de *iLO* se puede hacer lo siguiente:

- Arrancar o apagar remotamente el servidor.
- Monitorizar el estado de las interfaces de red, discos duros en uso, dispositivos conectados, ...
- Monitorizar la temperatura del servidor (desde varios sensores) y el estado de los ventiladores.
- Montar imágenes de disco remotamente.
- Configurar y leer *logs* de eventos.
- Utilizar herramientas de *troubleshooting*: tratamiento guiado de errores, notificaciones al fabricante, ...

Es posible proteger el acceso a *iLO* empleando un servidor *Kerberos* de autenticación ya existente en la organización, e incluso, servidores *LDAP* o *AD*. En esta misma línea, puede configurarse un acceso *SSO* (*Single*

Sign-On) desde una sesión ya abierta. La comunicación puede cifrarse mediante *AES* o *3DES*.

Respecto a las consultas *SNMP*, el funcionamiento de *iLO* es especial (conocido como *SNMP Pass-thru*): puede recibir consultas *SNMP*, pero *iLO* las pasará al sistema operativo en ejecución. La respuesta del agente (*HP Insight Agent*) instalado en el *SO* volverá a travesar el subsistema *iLO* para posteriormente ser devuelta al cliente. Analizaremos las posibilidades de este funcionamiento al desarrollar módulos específicos, llegando a instalar agentes software en los *SO* base si se considera necesario.

También es posible configurar *traps SNMP* para alertar de situaciones anómalas, aunque preferiremos evitar su uso siempre que no resulte estrictamente necesario.

Una forma alternativa de obtención de información es emplear *IPMI* (*Intelligent Platform Management Interface*). Esta interfaz, disponible solo para *iLO v3* o superior, es accesible a nivel de hardware, es decir, sin necesidad de interactuar con el *SO*. Está a la escucha del puerto 623.

Dado que *Pandora FMS* no es compatible nativamente con *IPMI*, estudiaremos las posibilidades de integración de *plug-ins* o extensiones que lo soporten solo si resulta estrictamente necesario (es decir, si alcanza información que no puede ser alcanzada mediante *SNMP* sin agente software). Preventivamente, instalamos el software libre *IPMITools* junto con todas sus dependencias en el servidor *Pandora FMS*.

4.3.1.4.- Estándares *WBEM* y *CIM*

Como se ha mencionado anteriormente, para monitorizar el *hardware* desde un sistema operativo genérico (*Windows*, *Linux*, *ESX*, ...), este proceso se reduciría a la instalación del agente *HP Insight Manager* sobre el mismo *SO*. Este agente estaría comunicado con el *iLO* del propio servidor, de modo que las consultas *SNMP* se realizarían desde *Pandora FMS* contra el propio *iLO* (en el modo *pass-thru* anteriormente descrito).

Sin embargo, los agentes para el *SO VMware ESXi* no permiten este funcionamiento, ya que *HP* y *VMware* consideran que el protocolo *SNMP* es inseguro (sin cifrar en sus versiones más comunes y sobre *UDP*). En su lugar⁸, proponen el uso del estándar *WBEM* (*Web Based Enterprise Management*), que a su vez se basa en el modelo de datos *CIM* (*Common Information Model*).

El modelo de datos *CIM* define una serie de estructuras (bajo el nombre de "*namespaces*") de la que dependen un conjunto de clases y subclases con unos determinados atributos.

⁸ *VMware* no se desprende completamente del protocolo *SNMP* en *ESXi*. De hecho, es posible activar este servicio sobre el puerto *UDP/161* habitual, aunque la información del *hardware* que otorga no nos resulta suficiente. Es posible que estas consultas *SNMP* sobre el propio *ESXi* resulten de utilidad en capítulos posteriores.

Por tanto, en cada consulta *WBEM* se deberá indicar el nombre de la clase y el *namespace* al que pertenece dentro del modelo *CIM*. Todas las comunicaciones entre un cliente *WBEM* y el servicio dentro de *ESXi* se realizan en formato *XML* sobre *HTTPS* en el puerto *TCP/5989*.

En general, nos resultaría interesante monitorizar:

- Estado de las fuentes de alimentación.
- Temperatura del servidor.
- Estado de dispositivos de almacenamiento: número de dispositivos conectados, uso de capacidad, ...
- Estado de *CPU*.
- Estado de los módulos de memoria *RAM*.
- Errores del sistema.
- ...

4.3.2.- Dispositivos de Prueba

Se nos otorga acceso a la plataforma *iLO* del siguiente servidor, que se encuentra actualmente en producción en un cliente de la empresa:

Servidor HP ProLiant DL380 G7

- *HP_ProLiant_DL380G7_1*

Por otra parte, la empresa nos facilita otro equipo, propiedad de un cliente pero no utilizado actualmente, para realizar instalaciones de agentes y evaluaciones de soluciones sin poner en riesgo la continuidad de negocio del cliente.

El modelo y el alias utilizado en *Pandora FMS* son:

Servidor HP ProLiant DL360 G5

- *HP_ProLiant_DL360G5_1*

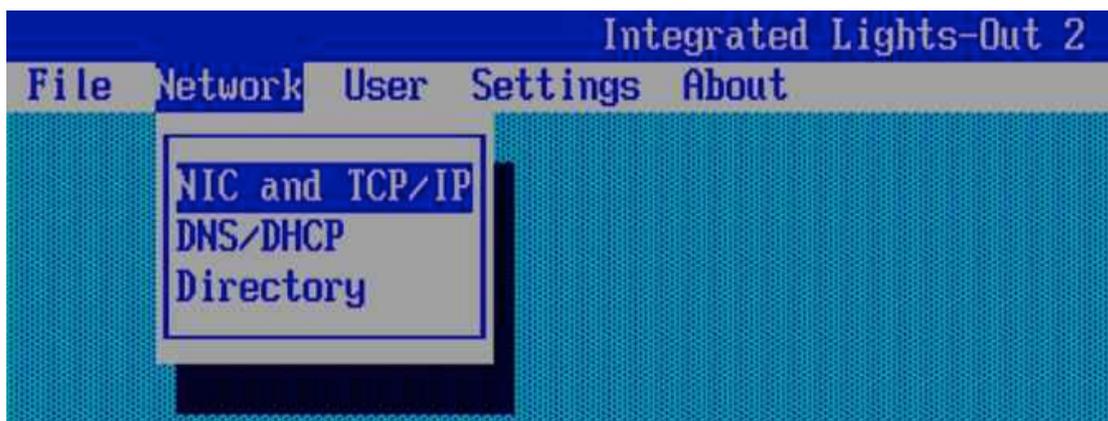
En el siguiente subapartado cubriremos la instalación y puesta en marcha de dicho servidor:

4.3.2.1.- Instalación y Puesta en Marcha de Servidor HP ProLiant DL360 G5

El servidor posee las siguientes características técnicas:

- 2 *CPUs Intel Xeon (Quad Core)* 2'33 GHz
- 8 GB *RAM*
- 2 *HDD* 40GB (en *RAID1*)
- *iLO2*
- *Hostname: esxi-pfg*

En primer lugar, arrancamos el equipo (sin sistema operativo) y accedemos al panel de control de *iLO2* para configurar su interfaz de red. Recordemos que, en los servidores *HP ProLiant*, se reserva una interfaz física de red para el subsistema *iLO*.

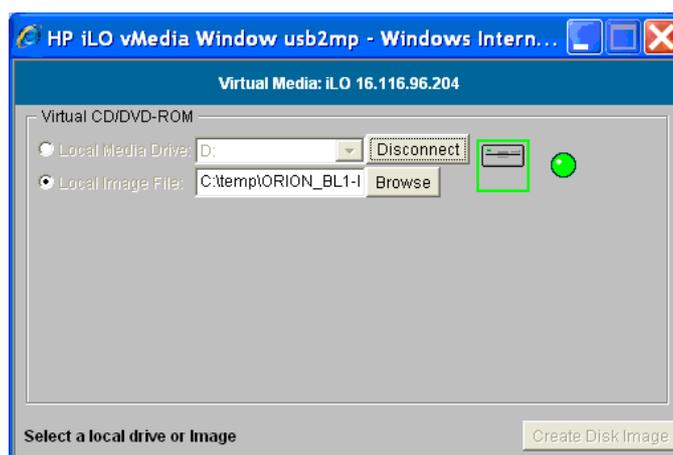


Una vez realizado este paso, seremos capaces de acceder a la interfaz web de *iLO* desde nuestro equipo personal (que se encuentra en la misma red de área local).

El siguiente paso es actualizar su *firmware*, para lo cual se debe descargar su imagen en extensión *.bin* y cargarla a través de la interfaz web de *iLO*. Esta actualización resulta necesaria para que la plataforma responda a *IPMI 2.0* (el empleado en consultas *IPMI-over-LAN*) y para posibilitar el montaje remoto de imágenes de disco *.iso*.

Concretamente, descargamos una imagen de disco del OS *VMware ESXi 5.1* personalizada para servidores *HP ProLiant*, que contendrá una serie de paquetes, *drivers* y agentes específicos ya instalados.

Cargamos esta imagen remotamente (desde una aplicación *Java* accesible vía web conocida como *Remote Console*) en un lector *CD/DVD* virtual y reiniciamos (también remotamente) el servidor:



A continuación, instalamos el sistema operativo desde el lector *CD/DVD* virtual de manera habitual. Tras instalar el *SO* y configurar su interfaz de red, instalamos el *VMware vSphere Client* y *VMware vSphere CLI* en nuestra máquina local. A partir de ahora, todas las labores de administración (creación de máquinas virtuales, arranque de servicios, etc.) se realizará a través de esta aplicación.

4.3.2.2.- Configuración de WBEM/CIM

El objetivo de esta sección es configurar el servidor y *Pandora FMS* de tal modo que seamos capaces de realizar la monitorización del *hardware* del servidor *ProLiant* a través de *WBEM*.

En primer lugar, nos aseguramos que se encuentran instalados los agentes de *HP* (conocidos como "*WBEM providers*"), que serán los encargados de recolectar la información del estado del *hardware* del servidor y dejarla disponible en una estructura basada en *CIM* (bajo los *namespaces* "*cimv2*" o "*hpq*", entre otros).

En segundo lugar, debemos instalar el cliente *WBEM* sobre la distribución *CentOS* que aloja el *Pandora FMS Server*. El único cliente existente es el módulo *PyWBEM* para *Python*.

Por tanto, para realizar consultas desde la plataforma de monitorización, se deberán desarrollar *scripts* en *Python* que empleen métodos de *PyWBEM* y que generen un *output* que pueda manipularse fácilmente desde *Pandora FMS*.

En este punto, generamos los *scripts* genéricos "*hp_cim_cimv2.py*" y "*hp_cim_hpq.py*" para realizar consultas a clases dentro de los *namespaces* *cimv2* y *hpq* respectivamente. Estos *scripts* serán los encargados de realizar las consultas y serán invocados desde otros *scripts* que formateen su *output* a medida para *Pandora*, por lo que decidimos situarlos en la ruta *"/usr/bin"*.

4.3.3.- Creación de Agentes

Desde la consola web de *Pandora FMS*, creamos los agentes *HP_ProLiant_DL380G7_1* y *HP_ProLiant_DL360G5_1*.

4.3.4.- Desarrollo de Módulos

En este apartado, distinguiremos los módulos en función del método de extracción de información: consultas vía *IPMI* o consultas *WBEM/CIM*. Dejaremos sin clasificar el clásico *ping*:

- **Host Alive:** Comprueba que la interfaz *iLO* se mantiene activa, mediante el comando *ping*.

4.3.4.1.- Vía IPMI

Aquí se definen los módulos generados para realizar consultas a través de *IPMI*. Es importante destacar que este tipo de chequeos no tienen por qué ser válidos para todas las versiones o generaciones de los servidores, y que los aquí listados han sido desarrollados específicamente para los modelos indicados por la empresa:

- **Chassis Power:** Devuelve el estado de la alimentación del chasis (*on/off*). Recordemos que la máquina puede estar apagada aún cuando la interfaz *iLO* responde a *ping*.

Este módulo realiza una consulta *IPMI*, por lo que se ha desarrollado el script "*hp_ipmi_chassis_power.sh*" y se ha configurado el servidor

de *plug-ins* de *Pandora FMS* (tal y como se hizo para módulos anteriores de pérdida de paquetes y uso de *PoE*) para poder realizar el chequeo.

Lo consideraremos *critical* si su valor es 0 (es decir, si su estado no es "ON").

- **Power Meter:** Consumo actual de potencia (en *wattios*) del servidor. De nuevo, se ha programado el *script* "*hp_ipmi_power_meter.sh*" y se ha realizado el proceso de creación de *plug-in* habitual.
- **Power Supply 1:** Potencia que entrega la fuente de alimentación #1, medida en *wattios*. Creamos otro *script* "*hp_ipmi_power_supply.sh*" y configuramos otro *plug-in*. Un valor de 0 indicará que no se está utilizando, por lo que se le asignará un estado *critical*.
- **Power Supply 2:** Ídem, reutilizando el *plug-in* del módulo anterior.
- **Temperatura Media CPU:** Mide la temperatura media de todos los sensores de *CPU* (sensores "*Temp 2*" y "*Temp 3*"), en °C. Para ello, se genera el *script* "*hp_ipmi_temp_cpu.sh*". Siguiendo las indicaciones del fabricante (mostradas en *iLO*), mostraremos *warning* con una temperatura superior a los 65°C y *critical* con una superior a los 83°C.
- **Temperatura Ambiental:** Ídem para la temperatura ambiental (sensor "*Temp 1*"). El *plug-in* utilizará el *script* "*hp_ipmi_temp_ambiental.sh*". *Warning* a 40°C y *critical* a 45°C.
- **Temperatura Media RAM:** Ídem para la memoria *RAM* (sensores "*Temp 4*", "*Temp 5*", "*Temp 6*" y "*Temp 7*"). *Script* "*hp_ipmi_temp_ram.sh*", con *warning* a los 70°C y *critical* a los 92°C.
- **Temperatura Almacenamiento:** Ídem para los dispositivos de almacenamiento (sensor "*Temp 29*"). *Script* "*hp_ipmi_temp_disk.sh*", siendo *warning* a los 55°C y *critical* a los 65°C.
- **Temperatura Media Power Supply:** Ídem para las fuentes de alimentación (sensores "*Temp 8*" y "*Temp 9*"). *Script* "*hp_ipmi_temp_fuentes.sh*", siendo *warning* a los 65°C y *critical* a los 70°C).
- **Temperatura Media Sistema:** Ídem para otros sensores del sistema ("*Temp 10*", "*Temp 11*", "*Temp 12*", "*Temp 19*", "*Temp 20*", "*Temp 21*", "*Temp 22*", "*Temp 23*", "*Temp 24*", "*Temp 25*", "*Temp 26*" y "*Temp 30*"). *Script* "*hp_ipmi_temp_sys.sh*", *warning* a los 70° y *critical* a los 75°C.
- **Ventilador N:** Muestra, en %, las *RPM* actuales de un ventilador de índice *N* (del 1 al 6) en relación con las *RPM* máximas que puede

alcanzar. Se emplea el script "*hp_ipmi_fan.sh*". Un valor de 0 o no numérico se alertará como *critical*, dando a entender que el ventilador no está funcionando correctamente y debe ser sustituido.

4.3.4.2.- Vía WBEM/CIM

***NOTA:** Aunque estos módulos se encuentren bajo el mismo agente, debemos indicar en ellos la dirección *IP* de la interfaz de red que utiliza *ESXi* (en lugar de la dirección *IP* de *iLO*).

- **Estado General:** Muestra el estado general del sistema, pudiendo ser "OK", "Degraded" o "Major Failure". Mediante "*hp_wbem_estado_general.sh*", se consulta la clase *CIM* que siempre indica el estado más crítico de algún componente *hardware* que esté detectando. Además de los estados mencionados, se mostrará *Warning* siempre que el script devuelva un valor distinto de *OK* que no sea *Major Failure*, mediante la expresión regular `^(?!OK).*`
- **RAM Instalada:** Muestra cuánta memoria *RAM* tiene instalada el servidor, medida en *GB*. Al considerarlo escaso, mostramos *warning* al poseer menos de 4 *GB* de memoria. Se emplea el script "*hp_wbem_ram_cap.sh*".
- **RAM Slots Ocupados:** Slots de memoria *RAM* que contienen algún módulo *RAM* operativo. Informativamente, mostraremos *warning* si se encuentran instalados 2 módulos o menos. Se emplea el script "*hp_wbem_ram_slots.sh*".
- **RAM Estado:** Muestra el estado conjunto de los módulos de memoria *RAM* instalados ("OK", "Degraded" o "Critical"). El estado de este módulo es el estado más grave de cada módulo de memoria individual. Para ello, se ha generado el script "*hp_wbem_ram_estado.sh*".
- **CPUs Estado:** Muestra el estado conjunto de los procesadores ("OK", "Degraded" o "Critical"). De nuevo, el estado de este módulo será el estado más grave de cada procesador individualmente. Se consulta mediante el script "*hp_wbem_cpu_estado.sh*".
- **Chasis Estado:** Análogamente, muestra el estado del chasis del sistema. El script generado es "*hp_wbem_chasis_estado.sh*".
- **NICs Estado:** Ídem para el conjunto de las tarjetas de red del servidor. El script empleado es "*hp_wbem_nic_estado.sh*".
- **Power Supplies Estado:** Ídem, mostrando "Degraded" si alguna de las fuentes se encuentra en estado "Degraded" o "Critical". Script "*hp_wbem_ps_estado.sh*".
- **Smart Array:** Consulta el estado de un *storage pool* (que pueden ser discos físicos individuales o un conjunto de ellos en *RAID*) del servidor objetivo, dado un valor de *slot*. Se utiliza el script

"*check_esxcli_hparrray*" obtenido desde el repositorio oficial de *plug-ins* para *Nagios*, que ha sido modificado para ajustarlo a *Pandora FMS* y para mejorar el deficiente procesamiento de parámetros original.

El *script* devuelve un *string* informativo donde se indica la versión de la controladora *HP Smart Array*, el identificador del volumen lógico, la versión *RAID* utilizada, el tamaño del volumen lógico y su estado ("OK", "WARNING", "CRITICAL").

Cabe destacar que éste no es un chequeo estrictamente basado en *WBEM*, aunque sí accede a las clases *CIM* a través de comandos *ESXCLI* instalados en el servidor *Pandora FMS*.

4.3.5.- Configuración de Alertas

En este caso, configuraremos una alerta específica que nos enviará un *email* cuando el chasis esté apagado y disparará otra acción que tratará de volver a poner el chasis en marcha.

Creamos el comando "*IPMItool Chassis Power On*", que será el encargado de lanzar la aplicación *IPMItool* para ordenar vía *IPMI* al subsistema *iLO* que arranque la máquina. Se indican macros del tipo "*_address_*", "*_field1_*" y demás que serán sustituidas por *Pandora FMS* por su valores correspondientes:

Name	IPMItool Chassis Power On
Command	ipmitool -H <i>_address_</i> -I lanplus -U <i>_field1_</i> -P <i>_field2_</i> -L <i>_field3_</i> power on
Description	Field 1: Username Field 2: Password Field 3: Nivel de Privilegio (USER, ADMINISTRATOR,)

Creamos la acción asociada al comando anterior, de nombre "*iLO Chassis Power On*", indicándole los parámetros "*_fieldN_*" al comando:

Name	iLO Chassis Power On
Group	Servers
Command	IPMItool Chassis Power On + Create Command
Threshold	0 seconds
Field 1	Administrador
Field 2	[REDACTED]
Field 3	ADMINISTRATOR

Y finalmente, creamos la alerta:

Agent	Module	Template	Actions	Status
HP_ProLiant_DL360G5_1	Chassis Power	Critical condition	iLO Chassis Power On (On 1 Threshold 60) ✗ Mail to ander,[...]s@gmail.com (On 1 Threshold 60) ✗ + Add action	■

4.3.6.- Evaluación de Soluciones

La evaluación de las soluciones de monitorización adoptadas podrá realizarse de una forma ligeramente más agresiva que en los apartados anteriores, gracias al equipo de prueba del que disponemos. Las evaluaciones realizadas son las siguientes:

4.3.6.1.- Contraste de Información con la Plataforma iLO3

Se ha contrastado la información que muestra el subsistema *iLO3* sobre fuentes de alimentación, ventiladores, sensores de temperatura, tarjetas de red y estado general, verificando que ésta es correcta:



Cabe destacar que, mediante *IPMI*, *WBEM* y *ESXCLI*, el alcance de la monitorización en *Pandora FMS* es mucho mayor que el que ofrece la plataforma *iLO* en sus versiones 2 y 3. Por ejemplo, *iLO* no es capaz de acceder a la controladora *Smart Array*⁹ ni al estado concreto de las tarjetas de red, por lo que un fallo en éstos puede desembocar en estados diferentes en *iLO* y en *Pandora*.

4.3.6.2.- Apagado del Chasis

Forzamos el apagado remoto del servidor a través de *iLO* (vía web), a fin de provocar que se dispare la alerta configurada anteriormente.

Inmediatamente observamos que el chasis se muestra apagado en *Pandora*:



Tras un minuto (el tiempo definido en la alerta), comprobamos que *Pandora FMS* ha gestionado automáticamente la situación de error y se ha encargado de devolver el sistema a su estado normal:



4.3.6.3.- Desconexión de una Fuente de Alimentación

Como se ha mencionado anteriormente, los servidores *HP ProLiant* analizados pueden alimentarse eléctricamente a través de 2 fuentes de alimentación independientes. En este estado de redundancia, *Pandora FMS* muestra:



⁹ Al parecer, la nueva versión 4 de *iLO* sí que tiene capacidad para acceder a ella.

NORMAL: OK

Retirando un enchufe (de la primera fuente de alimentación, en este caso), observamos que el sistema sigue en marcha con la fuente de alimentación restante y se muestra:

Power Supplies Estado	Muestra el estado de las fuentes de alimentación. Si se encu...		N/A - N/A	
-----------------------	---	---	-----------	---

WARNING: Degraded

4.3.6.4.- Desconexión de un Módulo de Memoria RAM

Comprobamos que los módulos de *Pandora FMS* que están monitorizando la memoria *RAM* funcionan correctamente, retirando un módulo de memoria *RAM* y observando su impacto sobre la plataforma de monitorización.

Inicialmente, la máquina cuenta con 6 módulos de memoria *RAM* y un total de 8GB:

RAM Estado	Muestra el estado conjunto de los módulos de memoria RAM (OK...		N/A - N/A	
RAM Instalada	Memoria RAM instalada		4/0 - N/A	8 GB
RAM Slots Ocupados	Slots de memoria RAM ocupados		2/0 - N/A	6

Tras retirar 2 módulos de 1 GB, observamos¹⁰:

RAM Estado	Muestra el estado conjunto de los módulos de memoria RAM (OK...		N/A - N/A	
RAM Instalada	Memoria RAM instalada		4/0 - N/A	6 GB
RAM Slots Ocupados	Slots de memoria RAM ocupados		2/0 - N/A	4

4.3.6.5.- Conexión de una Tarjeta de Red

El servidor *DL360G5* cuenta con 2 *NIC (Network Interface Card)* dedicadas al *host* y otra específica para el subsistema *iLO2*. En un principio, solo se ha empleado una de las dos tarjetas de red de propósito general, estando la otra desconectada. *Pandora FMS* lo muestra como *warning*:

NICs Estado	Muestra el estado conjunto de las NIC (OK, Degraded o Critic...		N/A - N/A	
-------------	---	---	-----------	---

Para comprobar que el módulo opera correctamente, conectamos otro latiguillo *ethernet* a la segunda *NIC*, que está mapeado en el *switch* de acceso de la empresa. En este caso, se muestra un estado correcto:

NICs Estado	Muestra el estado conjunto de las NIC (OK, Degraded o Critic...		N/A - N/A	
-------------	---	---	-----------	---

4.3.6.6.- Desconexión de un Disco Duro

El servidor cuenta con dos discos duros en redundancia *RAID1*, por lo que son extraíbles en caliente (*hot-swappable*). Retiramos un disco duro, sin

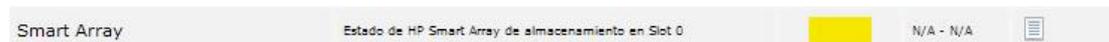
¹⁰ Aunque las prestaciones en cuanto a *RAM* se hayan reducido, el estado de ésta sigue siendo correcto. Nos resulta imposible forzar un error en alguno de los módulos (sin dañar el *hardware*, al menos), tal y como sucede en la monitorización del estado de las *CPUs*.

necesidad de apagar el chasis. Una vez repetido el chequeo *Smart Array*, *Pandora FMS* muestra:



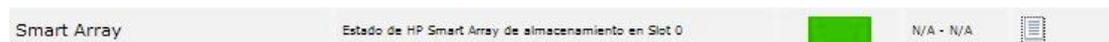
RAID CRITICAL - (Smart Array P400i in Slot 0 (Embedded) array A logicaldrive 1 (33,9 GB, RAID 1, Interim Recovery Mode))

Volvemos a insertar el disco en caliente. Mientras se recupera el estado normal de los discos en *RAID1* (actualizando el disco recién insertado), se muestra:



RAID WARNING - (Smart Array P400i in Slot 0 array A logicaldrive 1 (33,9 GB, RAID 1, Recovering 33% complete))

Finalmente, una vez se ha restablecido completamente la redundancia en espejo, observamos el estado correcto del módulo de *Pandora FMS* y su *output*:



RAID OK - (Smart Array P400i in Slot 0 (Embedded) array A logicaldrive 1 (33,9 GB, RAID 1, OK))

4.4.- Sistemas de Almacenamiento EMC²

4.4.1.- Análisis de Dispositivos

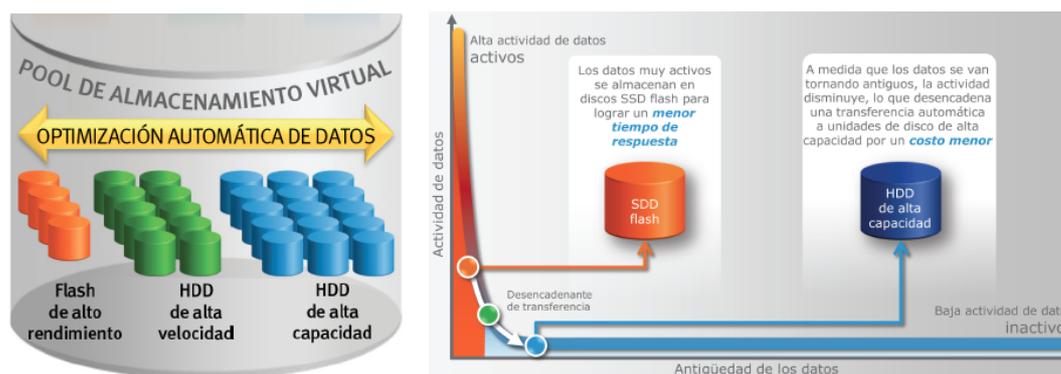
La firma EMC² ofrece una gran cantidad de soluciones de almacenamiento orientadas a entornos empresariales. En la empresa se contempla una familia de dispositivos de almacenamiento, que se divide en VNX (dispositivos de gama alta) y VNXe (dispositivos de gama media).

4.4.1.1.- VNX Series

La serie VMX de EMC² agrupa las soluciones de almacenamiento de altas prestaciones, llegando a ofrecer productos en formato de armario con capacidad para 1000 drives¹¹ y hasta casi 3 petabytes:



Para ofrecer unos mejores ratios de transferencia, los dispositivos VNX emplean distintos tipos de drives, diferenciados en tres niveles: un primer nivel de memoria de estado sólido (SSD), un segundo nivel de HDD rápidos y un tercer nivel de HDD más lentos pero de mayor capacidad.



Los datos fluyen dinámicamente entre los drives en función de su frecuencia de uso. Así, los datos manipulados más frecuentemente se hallarán en las memorias flash de alto rendimiento, e irán bajando de nivel a medida que vayan cogiendo antigüedad (sin ser utilizados). Este protocolo de caché

¹¹ Dado que los términos "disco" y "dispositivo" pueden resultar confusos e imprecisos, en lo sucesivo, se utilizará el término "drive" para hacer referencia a un dispositivo de almacenamiento genérico, que podrá ser HDD (Hard Drive Disk) o SSD (Solid State Drive).

recibe el nombre de *EMC FAST VP (Fully Automated Storage Tiering for Virtual Pools)*.

Soporta tanto el almacenamiento a nivel de fichero (mediante los protocolos *NFS* y *SMB/CIFS*) como el almacenamiento a nivel de bloque, accediendo mediante *iSCSI (Internet Small Computer System Interface)*, *FC (Fibre Channel)* o *FCoE (Fibre Channel over Ethernet)*. Por tanto, puede integrar una red *SAN (Storage Area Network)*, alojar un sistema *NAS (Network Attached Storage)* o una combinación de ambos.

Cuenta con potentes procesadores *Intel Xeon* (habituales en servidores convencionales) para cubrir las necesidades de procesamiento de las transacciones de datos, e incluso para instalar un sistema operativo sobre el dispositivo (necesario, entre otras cosas, para dar soporte a un sistema *NAS*). Estos subsistemas vienen a ser pequeños servidores tradicionales, al contar con procesador, memoria *RAM* y un pequeño disco *SSD* propio.

En cuanto a la tolerancia a fallos, estos dispositivos cuentan con fuentes de alimentación redundantes y baterías de respaldo, que permiten un apagado ordenado y el vaciado de caché ante un corte eléctrico. Los *drives* pueden funcionar en distintos modos de operación *RAID* y pueden ser extraídos en caliente. También cuentan con procesadores y conexiones de respaldo.

4.4.1.2.- *VNXe Series*

La serie *VNXe* ofrece unas prestaciones más limitadas y se ajusta a presupuestos más reducidos que su hermana mayor, la *VNX*. Responde a necesidades de almacenamiento que no exijan un rendimiento o una capacidad muy elevados, como podría ser el caso de gran parte de las *PYMEs*. Sin embargo, los dispositivos de la serie *VNXe* pueden ofrecer la no despreciable capacidad de hasta 150 *drives* y 450 TB.



También cuenta con conexiones, fuentes de alimentación y baterías redundantes, si bien no cuentan con protocolos de fibra (*FC* o *FCoE*). Pueden operar en *SAN*, *NAS* o una combinación de ambos, así como bajo los distintos modos *RAID*.

La tecnología de cacheado *EMC FAST VP* no está presente en los dispositivos de esta serie. Al igual que en *VNX*, cuenta con uno o dos pequeños servidores (denominados *SPs* o *Storage Processors*) que contienen procesadores *multicore Intel* de distintas gamas, memoria *RAM* dedicada y un pequeño *drive SSD*.

4.4.1.3.- *Unisphere Manager*

*EMC*² posee un entorno de gestión de almacenamiento que actualmente se incluye con todos los dispositivos de las series *VNX* y *VNXe*: el *Unisphere Manager*. Surge como una versión renovada del antiguo *Navisphere* y ofrece, entre otras cosas, una interfaz web desde la cual es posible monitorizar el estado del dispositivo y configurar sus distintos parámetros.

Lo que resulta de mayor interés para la monitorización remota es el *Unisphere CLI*, contenido en un paquete instalable para plataformas *Windows* y *Linux*. Esto nos permitirá crear nuestros scripts personalizados dentro del *Pandora FMS Server* para hacer consultas remotas (y, si procediera, otras acciones o reconfiguraciones).

Estos dispositivos también soportan el protocolo *SNMP*, únicamente mediante *traps*. Tal y como sucedía para otros dispositivos, y dada la escasa fiabilidad de las alertas mediante *traps*, evitaremos su uso en la medida de lo posible.

A través de consultas vía *CLI*, nos resultaría interesante monitorizar:

- Discos instalados, su estado y su uso.
- Estado de *RAID* y uso de *storage pools*.
- Estado de las fuentes de alimentación.
- Estado de las baterías.
- Estado de puertos y conexiones.

4.4.2.- *Dispositivos de Prueba*

Para acceder a los dispositivos de prueba, instalamos el *Unisphere CLI* en nuestro servidor *CentOS* que aloja al *Pandora FMS Server*. Este *software* está disponible en el sitio de descargas oficial de *EMC* en formato *RPM*.

Dado que emplea el protocolo *HTTPS* (sobre el puerto *TCP/443*) para establecer la conexión, el dispositivo tratará de autenticarse enviando un certificado digital propio. En cada consulta que realicemos mediante *Unisphere CLI*, se comprobará que éste certificado es válido o que al menos el cliente lo reconoce como tal.

Esta medida de seguridad supone que las consultas a nuevos equipamientos *VNX* o *VNXe* mediante *Unisphere CLI* no funcionarán hasta que hayamos instalado sus respectivos certificados digitales en la máquina de *Pandora FMS Server*.

Se nos proporciona acceso al siguiente equipo, mediante el *Unisphere Manager*, con rol de operador:

Cabina *VNXe 3100*:

- *EMC_VNXe3100_1*

4.4.3.- *Creación de Agentes*

Asociado al dispositivo de prueba, creamos el agente *EMC_VNXe3100_1*.

4.4.4.- Desarrollo de Módulos

Para desarrollar los módulos basados en consultas vía *Unisphere CLI* que procesen datos relativos al estado de cualquier subsistema (*drives*, baterías, fuentes de alimentación, ...), se han seguido unas pautas de diseño comunes.

Unisphere CLI contempla las siguientes relaciones entre el valor consultado (*health state*) y su significado:

Health State	Significado
5	OK
7	O.K. BUT
10	Warning/Degraded
15	Minor failure
16*	Unknown
20	Major failure
25	Critical failure
30	Non-recoverable error

* En realidad, para determinar un estado desconocido o *unknown* se emplea el valor 0 de *health state*. Decidimos introducir este valor en el conjunto de los "Warning", y por comodidad, sustituimos el valor 0 por el valor 16 dentro de los *scripts* generados. El valor numérico de *Health State* no se muestra en ningún momento fuera del *script*, por lo que esta consideración no tendrá mayores consecuencias.

Todos estos *scripts* tratarán los valores de *Health State* y generarán un *output* por salida estándar que contendrá el significado de dicho valor (por ejemplo, una dato de entrada de 20 generará el *output* "CRITICAL: Major failure").

Pandora FMS procesará este *output* para determinar el estado del módulo, mediante expresiones regulares: ".*CRITICAL.*" para filtrar los valores críticos y "^((?!OK).)*\$" para filtrar los valores de advertencia. Únicamente la cadena "OK" dará lugar a un estado correcto.

La lista de módulos generados es la siguiente:

- **Host Alive:** El clásico ping. Es el único módulo de estos dispositivos que no emplea comandos *CLI*.
- **Fuentes de Alimentación:** Muestra el estado más crítico de las fuentes de alimentación, en formato *string*. Este *output* se procesa tal y como se expresa anteriormente. El *script* utilizado es "emc_uemcli_ps_estado.sh".
- **Baterías:** Ídem para las baterías, con el *script* "emc_uemcli_bat_estado.sh".

- **Storage Pools Estado:** Ídem para los *storage pools*, con el script "*emc_uemcli_pools_estado.sh*"
- **Storage Pool Uso:** Muestra el uso (%) del espacio de almacenamiento disponible para un *storage pool* dado (por ejemplo, el de nombre "*Raid5_01*"). Será *warning* al superar el 80% y *critical* al superar el 90%. Se utiliza el script "*emc_uemcli_pool_uso.sh*".
- **Storage Pool Info:** Muestra información acerca del *storage pool* indicado (nombre, capacidad total, número de *drives*, modo *RAID*, ...). Se trata de un módulo meramente informativo, por lo que nunca mostrará *warning* ni *critical* (los aspectos concretos que sí generarían alertas ya han sido descompuestos en distintos módulos). Se emplea el script "*emc_uemcli_pool_info.sh*".
- **Discos Estado:** Muestra el estado más crítico entre todos los discos duros del sistema, en función de su valor de "*Health State*". Se utiliza el script "*emc_uemcli_discos_estado.sh*".
- **Módulos de Memoria Estado:** Ídem para los módulos de memoria *RAM* del sistema, en cada *SP*. Script "*emc_uemcli_mm_estado.sh*".
- **SSD Estado:** Ídem para los drives *SSD* presentes en los *Storage Processors*. Script "*emc_uemcli_ssd_estado.sh*".
- **SPs Estado:** Ídem para los *SPs* (*Storage Processors*). Script "*emc_uemcli_sp_estado.sh*".
- **Puerto Estado:** Ídem para los puertos de red. Puede indicarse un puerto concreto o un conjunto de ellos. Script "*emc_uemcli_port_estado.sh*".
- **Shared Folder Uso:** Muestra el uso (%) del espacio de almacenamiento de una carpeta compartida mediante *NFS* o *Samba/CIFS*. Script "*emc_uemcli_nas_uso.sh*".
- **Shared Folder Info:** Muestra información (*health state*, descripción, protocolo, servidor, memoria...) asociada a una carpeta compartida concreta. Script "*emc_uemcli_nas_info.sh*".
- **Snapshots Info:** Muestra información (estado, tipo de recurso, *LUNs* asociadas, etcétera) de todos los *snapshots* presentes en el sistema. Si no existe ningún *snapshot*, se mostrará *critical*. Se utiliza el script "*emc_uemcli_snaps_info.sh*".
- **Sesión de Replicación Info:** Muestra información acerca de la sesión de replicación especificada, como el tipo de consistencia, su *health state*, la última sincronización, el % de realización de la sincronización

(si se encuentra activa), el tipo de sesión (*iSCSI* o *Net*), etcétera. Script "*emc_uemcli_rep_info.sh*".

- **Alertas (Hoy):** Muestra las alertas sucedidas en el mismo día en el que se realiza el chequeo (a partir de las 00:00), tal y como se muestran en la interfaz web de la cabina. Se ignoran las alertas informativas, mientras que las marcadas como *critical* en la cabina también se marcan como *critical* en *Pandora*. Se mostrará *OK* si no existen nuevas alertas y *warning* para cualquier otro estado.

El script utilizado es "*emc_uemcli_alertas.sh*". Es posible indicar una antigüedad máxima de días para filtrar las alertas, para mostrar un abanico más amplio de ellas (de la última semana o el último mes, por ejemplo).

4.4.5.- Configuración de Alertas

Dada nuestra escasa capacidad para alterar el estado de los componentes de la cabina *VNXe*, no configuraremos ninguna alerta en *Pandora FMS*.

4.4.6.- Evaluación de Soluciones

Nuestra única fuente de información fiable es la interfaz web (*EMC Unisphere Web Console*), por lo que contrastaremos toda la información recolectada mediante *Unisphere CLI* con la mostrada en dicha interfaz.

4.4.6.1.- Contraste con EMC Unisphere Web Console

Tal y como hemos mencionado anteriormente, contrastamos la información de absolutamente todos los módulos con los mostrados en la *EMC Unisphere Web Console* (al fin y al cabo, la fuente básica de información en nuestras consultas mediante "*uemcli*" y en la interfaz web es la misma).

Esta interfaz web muestra la información gráficamente y la plasma sobre el mapa real de la cabina, lo que la convierte en una herramienta interesante para ser consultada una vez *Pandora FMS* haya alertado de una situación de error¹².

La interfaz tiene el siguiente aspecto:

¹² Una vez más, descubrimos que *Pandora FMS* no puede competir con las herramientas específicas de gestión en lo referente a la cantidad y calidad de presentación de información ofrecida. El rol de *Pandora FMS* siempre será el de alertar de situaciones anómalas y de ofrecer una reducida cantidad de información asociada, aunque una vez alertadas puede resultar conveniente acceder a las herramientas específicas para profundizar en la investigación del error y de sus causas.

The screenshot displays a web-based system health monitoring interface. At the top, there is a navigation bar with tabs for Dashboard, System, Storage, Settings, Hosts, and Support. The main content area is titled 'System Health' and is divided into several sections:

- System Info:** Displays details for the system named 'vnx_e_1', including its model 'VNXe3100', Product ID / SN 'CKM00111300024', and Software Version '2.2.0.17384'.
- System Components:** A tree view showing the status of various components. All components are marked with a green checkmark, indicating they are healthy. The components listed are: DPE, SP A, SP B, DPE PS B, DPE PS A (highlighted), DPE Disk 0 through DPE Disk 11.
- Power Supply: DPE PS A:** A large image showing the physical power supply unit installed in the server rack. A green box highlights the unit.
- Component Description:** Provides technical details for the selected power supply:
 - Type: Power Supply
 - Slot: 0
 - Product ID / SN: ASTAB110900153
 - Part Number: 071-000-537

4.5.- Entorno de Virtualización VMware

4.5.1.- Análisis del Entorno

Las soluciones de virtualización surgen como alternativa a la instalación de sistemas operativos tradicionales (*Windows Server, RHEL, SLES, Solaris, ...*) sobre servidores físicos.

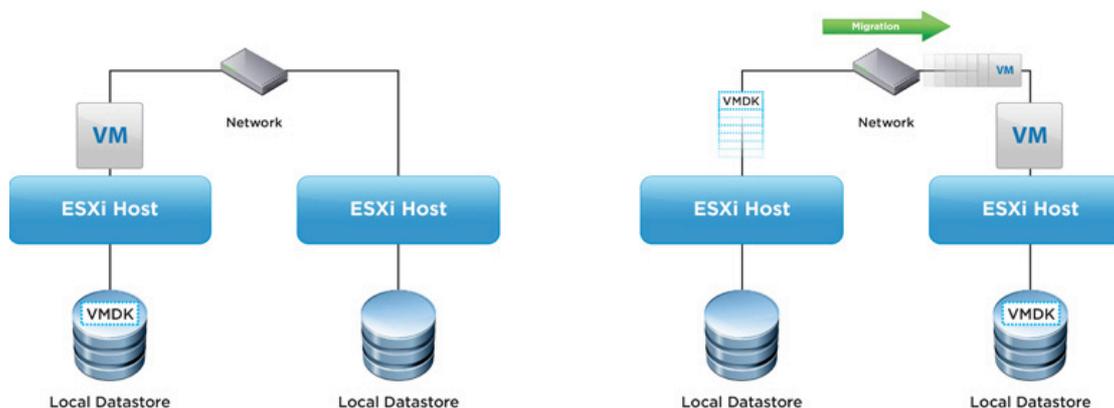
Un mismo entorno de virtualización (como *VMware ESX* y *ESXi*, que estudiaremos a continuación) permite alojar varios *SO* tradicionales e instalarlos/desinstalarlos a través de una interfaz única. Su principal ventaja es la flexibilidad y facilidad de gestión: frente a la instalación de equipos físicos, se tiene una infraestructura en la que los recursos (memoria, *CPU*, interfaces de red, etc) se asignan dinámicamente a cada *SO* alojado.

Esto también viene a simplificar el *CPD* de la empresa: en lugar de multitud de máquinas simples, que cuentan con procesador, memoria y discos independientes, se opta por una única máquina mucho más potente en cuanto a recursos¹³ (contando con varios procesadores, grandes cantidades de memoria, múltiples slots para discos duros, ...).

4.5.1.1.- Tecnología vMotion y High Availability (HA)

Una característica necesaria en servidores en producción es la alta disponibilidad (más conocida por sus siglas en inglés: *HA, High Availability*). Esto supone contar con la redundancia de aquellos elementos que sean críticos para el funcionamiento normal del sistema: fuentes de alimentación, discos duros, tarjetas de red, ...

En este aspecto, *VMware* cuenta con una tecnología de balanceo de carga dinámico que resulta útil para estos fines: *vMotion*.



Esta tecnología permite combinar varias instancias de *VMware ESX* o *ESXi* en varios servidores físicos intercomunicados en una misma LAN. Cuando un servidor se encuentre al límite de carga o cuando sea necesario detener un

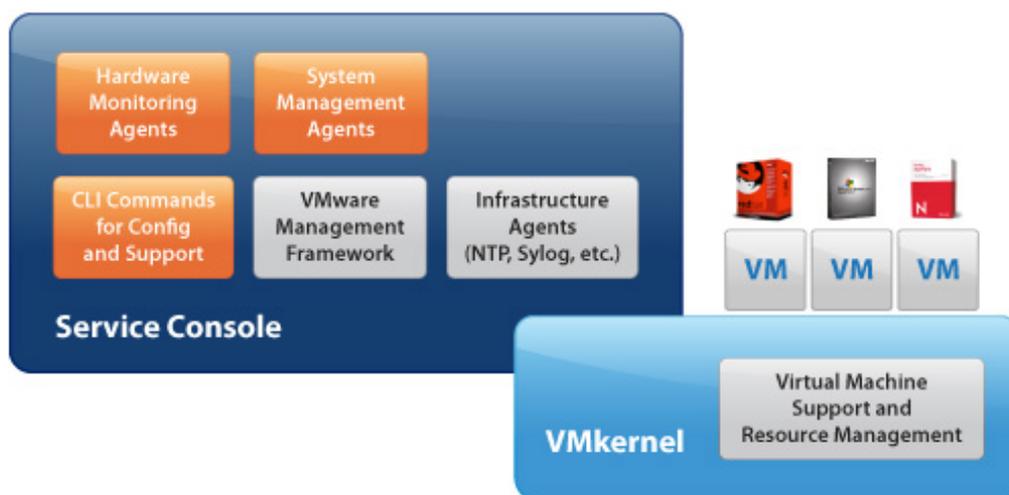
¹³ Véase el apartado de *Servidores HP ProLiant*. Como curiosidad, algunos de los modelos más recientes (*DL980 G7*) pueden contener hasta 8 procesadores de 10 núcleos y 2 terabytes de memoria *RAM*.

servidor por labores de mantenimiento, *vMotion* migrará una o varias máquinas virtuales de un servidor a otro a través de la red (puerto *TCP/8000*) y de manera totalmente transparente al usuario.

Para realizar la migración de una máquina virtual, esta se copiará en el servidor objetivo mientras la *VM* sigue activa. Una vez copiada, se transferirá el estado de ejecución concreto, se suspenderá la *VM* en el servidor original y se lanzará en el servidor objetivo. *VMware* estima que, en una red *Gigabit Ethernet*, el tiempo de interrupción del servicio es inferior a 2 segundos.

4.5.1.2.- VMware ESX

VMware ESX constituye el sistema operativo clásico sobre el que se montan directamente multitud de máquinas virtuales. Emplea un sistema *Linux* basado en *Red Hat Enterprise Linux (RHEL)* modificado para permitir la ejecución del hipervisor (del tipo "*bare metal*", es decir, montado sobre el mismo *kernel* del *SO* y en comunicación directa con el *hardware*) y otros componentes de virtualización de *VMware*.

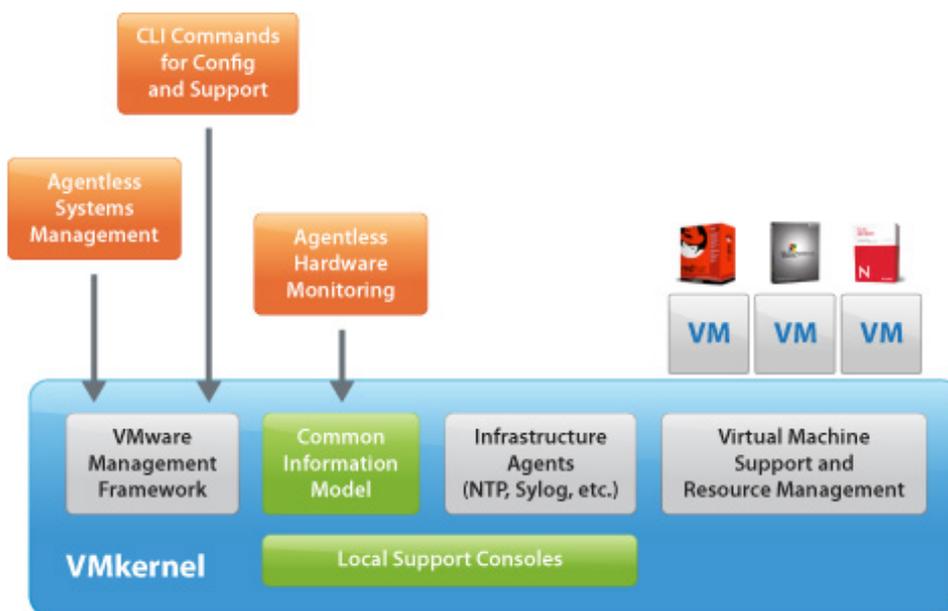


La capa *Linux* (conocida como consola o *Service Console*) permite instalar fácilmente agentes para monitorizar y gestionar el sistema completo.

No obstante, *VMware* ha dejado atrás toda la línea *ESX* en favor de los nuevos sistemas *ESXi*. Por ello, no diseñaremos soluciones de monitorización para sistemas *ESX*.

4.5.1.3.- VMware ESXi

Los nuevos sistemas *VMware ESXi* introducen un cambio drástico en la antigua arquitectura *ESX*, pretendiendo optimizar el rendimiento del entorno de virtualización.



El cambio más notable es la eliminación de la *Service Console* que alojaba distintos agentes de terceros. A cambio, el *kernel* del sistema pasa a tener un mayor protagonismo al encargarse de soportar el hipervisor y ciertos agentes autorizados (como los *CIM Providers* de HP, analizados en el capítulo de monitorización física de estos servidores).

En este capítulo, nuestro interés recae sobre los aspectos del sistema y no sobre el estado del *hardware* (ya analizado). Si bien la monitorización del *hardware* requería de *WBEM* y *CIM*, esta vez nos bastará con consultas *SNMP* sobre el propio servicio *snmpd* de *ESXi* en la mayoría de los casos.

En resumen, los aspectos que nos interesaría monitorizar sobre *ESXi* son:

- Número de *VMs* activas, suspendidas o apagadas.
- Estado de servicios o de agentes.
- Uso de memoria, *CPU*, red, ...
- ...

4.5.2.- Entornos de Prueba

Contaremos con acceso a una instancia de VMware ESXi 5.1 montada sobre el servidor *HP ProLiant* de prueba utilizado en el capítulo de monitorización de hardware de estos servidores.

Sistema VMware ESXi 5.1:

- *VMware_ESXi_5.1_1*

4.5.2.1.- Activación de *snmpd* sobre ESXi 5.1

Dado que ya se ha cubierto la instalación del sistema, procedemos directamente a activar el servicio *SNMP* sobre el mismo.

En primer lugar, desde el cliente *vSphere*, habilitamos el servicio *SSH* para poder acceder al *ESXi-CLI* (línea de comandos) remotamente.

Una vez hemos accedido, modificamos el fichero:

```
/etc/vmware/snmp.xml
```

Habilitamos *SNMP* e introducimos la comunidad de consulta. Posteriormente arrancamos el servicio:

```
/etc/init.d/snmpd start
```

4.5.2.2.- Instalación de Máquinas Virtuales

Instalaremos dos máquinas virtuales sobre el sistema, a fin de obtener más fuentes de datos y poder realizar comprobaciones de manera más efectiva.

A través del cliente *vSphere*, cargamos un *CD* virtual con la imagen "*CentOS-6.4-i386-minimal.iso*" que la máquina virtual detectará como si el *CD* estuviera introducido físicamente en el servidor.

A partir de esta imagen de disco, creamos dos máquinas virtuales *CentOS_1* y *CentOS_2* con características muy reducidas (ya que nos resulta indiferente) e idénticas:

- *CentOS 6.4 Minimal* de 32 bits.
- 1 procesador virtual.
- 1 GB de memoria *RAM*.
- 8 GB de espacio en disco.

4.5.3.- Creación de Agentes

Creamos el agente *VMware_ESXi5.1_1* asociado al sistema operativo de prueba que tenemos disponible.

4.5.4.- Desarrollo de Módulos

Desarrollamos los siguientes módulos:

- **Host Alive**
- **VMs Encendidas:** Muestra el número de máquinas virtuales arrancadas en el sistema. Se desarrolla el *script* "*esxi_vms_estado*", empleado en este caso con la opción "*-O Encendidas*" para mostrar el valor descrito. Dado que es un módulo meramente informativo, no mostraremos estados *warning* ni *critical*.
- **VMs Apagadas:** Ídem para las máquinas virtuales instaladas pero no arrancadas en el sistema. Se utiliza la opción "*-O Apagadas*" del *script* "*esxi_vms_estado*".
- **VMs Totales:** Ídem para las máquinas virtuales instaladas pero no arrancadas en el sistema. Se utiliza la opción "*-O Totales*" del *script* "*esxi_vms_estado*".

- **RAM Uso:** Uso actual de la memoria *RAM* por el sistema *ESXi* completo. Utiliza el *script* "*esxi_mem_uso.sh*" con la opción "*-O RAM*". Muestra *warning* al alcanzar el 80% de uso y *critical* al superar el 90%.
- **Espacio de Disco Uso:** Ídem para el espacio de disco disponible en la partición principal *VMFS*. Utiliza el *script* "*esxi_mem_uso.sh*" con la opción "*-O RAM*". Muestra *warning* al alcanzar el 80% de uso y *critical* al superar el 90%.
- **RAM Allocation:** Muestra la relación (%) entre la memoria *RAM* reservada para las *VMs* arrancadas en el sistema y la memoria *RAM* total disponible. Consideraremos *warning* todo aquel valor superior al 100%.

Se considera un indicador interesante que alerta de una posible saturación futura, ya que la suma de la *RAM* reservada para las máquinas virtuales excede la capacidad real del sistema. Para realizar esta consulta se ha desarrollado el *script* "*esxi_ram_allocation.sh*".

- **NIC #1 Tráfico Bajada:** Tráfico de bajada (bits/s) a través de una tarjeta de red del sistema (en este caso es la # 1, la única activa del servidor). Nótese que este módulo mide el tráfico conjunto de todo el sistema, incluyendo las máquinas virtuales. Esta consulta se realiza mediante una consulta *SNMP* simple a la tabla *ifTable*.
- **NIC #1 Tráfico Subida:** Ídem para el tráfico de subida.
- **CPU Uso:** Muestra el uso de *CPU* medio (entre todos los procesadores) del sistema, en %. Consideraremos *warning* los valores superiores al 70% y *critical* los superiores al 90%.

Aunque el módulo no resulte aparentemente complejo, es necesario establecer una sesión *SSH* con el servidor *ESXi* y ejecutar la lógica del *script* "*esxi_cpu_uso.sh*" remotamente (en el propio *ESXi*).

- **Latencia DriverRead:** Muestra el tiempo (en ms) que tarda el *driver* de disco en tramitar una orden de lectura, sin que ésta llegue a completarse. Es decir, indica el tiempo transcurrido desde que el *kernel* de *ESXi* envía la petición de lectura al *driver* hasta que el *driver* transmite dicha orden al dispositivo físico.

VMware indica que un valor superior a 15 ms (tanto en lectura como en escritura) indica un error de *driver* o una saturación del mismo (por ejemplo, si las máquinas virtuales realizan demasiadas operaciones sobre el disco conjuntamente). Por ello, establecemos un valor de *warning* de 10 a 15 ms y *critical* para valores superiores.

Se emplea el *script* "*esxi_disk_iostats.sh*", que emplea el servicio *SSH* para recabar la información.

- **Latencia *DriverWrite***: Ídem, para operaciones de escritura en disco.
- **Latencia *KernelRead***: Ídem, para el tiempo que el *kernel* tarda en transferir una petición de escritura al *driver*. En esta ocasión, *VMware* indica que los tiempos rara vez son superiores al milisegundo, por lo que mostraremos *warning* entre 1.5 y 3 ms y *critical* para valores superiores.
- **Latencia *KernelWrite***: Ídem, para operaciones de escritura en disco.

4.5.5.- Configuración de Alertas

Vistas las alertas de los apartados anteriores, no se considera desarrollar ninguna nueva alerta que resulte de especial interés.

4.5.6.- Evaluación de Soluciones

Realizamos las siguientes pruebas, a fin de comprobar que los módulos operan según lo establecido:

4.5.6.1.- Apagado y Encendido de VMs

Observamos el estado inicial, en el que la *VM CentOS_1* se encuentra apagada y *CentOS_2* encendida:

VMs Apagadas	Muestra el número de VMs instaladas pero apagadas en el sist...		N/A - N/A	1
VMs Encendidas	Muestra el número de VMs encendidas en el sistema		N/A - N/A	1
VMs Totales	Consulta el número de VMs totales instaladas en el sistema		N/A - N/A	2

Los indicadores de memoria se muestran correctos:

Espacio de Disco Uso	Muestra el uso (%) del espacio de almacenamiento disponible		90.0/80 - 0/90	62.8 %
RAM Allocation	Muestra el % entre la memoria RAM reservada en las VMs arran...		0/100 - N/A	12.8 %
RAM Uso	Uso de memoria RAM (%) por parte de ESXi		90.0/80 - 0/90	25.5 %

Arrancamos *CentOS_1* y observamos el cambio:

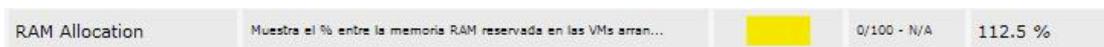
VMs Apagadas	Muestra el número de VMs instaladas pero apagadas en el sist...		N/A - N/A	0
VMs Encendidas	Muestra el número de VMs encendidas en el sistema		N/A - N/A	2
VMs Totales	Consulta el número de VMs totales instaladas en el sistema		N/A - N/A	2

También observamos cómo se resienten los porcentajes de uso de la memoria *RAM* y de disco:

Espacio de Disco Uso	Muestra el uso (%) del espacio de almacenamiento disponible		90.0/80 - 0/90	66.6 %
RAM Allocation	Muestra el % entre la memoria RAM reservada en las VMs arran...		0/100 - N/A	25.3 %
RAM Uso	Uso de memoria RAM (%) por parte de ESXi		90.0/80 - 0/90	28.5 %

4.5.6.2.- Exceso de Reserva de Memoria RAM

Desde el cliente *vSphere*, reservamos 8GB de memoria *RAM* para la máquina *CentOS_2*. Tener arrancadas las dos *VMs* supone ahora una reserva de 9GB de memoria *RAM*, superior a los 8GB que posee el servidor:



4.5.6.3.- Contraste con vSphere Client

Comprobamos que la información otorgada por los módulos de memoria *RAM*, disco, uso de *CPU* y tráfico de red concuerdan con los mostrados en la plataforma *vSphere*, que se presenta del siguiente modo:



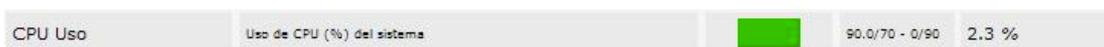
4.5.6.4.- CPU Stress Test

Realizaremos un test de estrés sobre una máquina virtual, a fin de observar que el módulo "*CPU Uso*" refleja correctamente situaciones de carga excesiva de *CPU*.

En primer lugar, para que el test de estrés sea totalmente efectivo, reconfiguramos *CentOS_1* para que pueda utilizar todos los núcleos de procesador disponibles (*2 CPUs Quad Core*).

Aunque existen alternativas "caseras", nos decantamos por instalar el paquete "*stress*", especializada en realizar tesis de estrés como el que queremos en esta sección.

Partimos de un uso de *CPU* mínimo:



Durante la realización del *stress test*, observamos que el uso se eleva hasta más del 88% :

```
[root@centos1 ~]# stress --cpu 8 --timeout 60s
stress: info: [1330] dispatching hogs: 8 cpu, 0 io, 0 vm, 0 hdd
```

CPU Uso	Uso de CPU (%) del sistema		90.0/70 - 0/90	88,7 %
---------	----------------------------	--	----------------	--------

Por último, comprobamos el efecto de limitar la máquina virtual a 4 *cores* realizando exactamente el mismo test. Se observa que, en estas condiciones, un pico en el consumo de *CPU* no lastra completamente el sistema:

CPU Uso	Uso de CPU (%) del sistema		90.0/70 - 0/90	50,9 %
---------	----------------------------	--	----------------	--------

4.5.6.5.- Disk Stress Test

Mediante la misma herramienta, ponemos a prueba los indicadores del rendimiento de lectura/escritura en disco. Inicialmente observamos que sus valores se encuentran en rangos que se ajustan a lo esperado, en una situación de baja carga:

Latencia DriverRead	Muestra el tiempo (en ms) que tarda el driver de disco en tr...		15/10 - 0/15	2,6 ms
Latencia DriverWrite	Muestra el tiempo (en ms) que tarda el driver de disco en tr...		15/10 - 0/15	5,0 ms
Latencia KernelRead	Muestra el tiempo (en ms) que tarda el kernel en tramitar un...		3/1.5 - 0/3	0,0 ms
Latencia KernelWrite	Muestra el tiempo (en ms) que tarda el kernel en tramitar un...		3/1.5 - 0/3	0,0 ms

Lanzamos el test de estrés sobre el disco, que lo forzará a realizar una gran carga de trabajo en operaciones de lectura y escritura:

```
[root@centos1 ~]# stress --io 8 --hdd 8 --hdd-bytes 1GB --timeout 300s
stress: info: [1456] dispatching hogs: 0 cpu, 8 io, 0 vm, 8 hdd
```

Observamos que, efectivamente, los módulos de *Pandora FMS* reflejan este exceso de carga y han rebasado el nivel de alerta establecido:

Latencia DriverRead	Muestra el tiempo (en ms) que tarda el driver de disco en tr...		15/10 - 0/15	20,9 ms
Latencia DriverWrite	Muestra el tiempo (en ms) que tarda el driver de disco en tr...		15/10 - 0/15	19,1 ms
Latencia KernelRead	Muestra el tiempo (en ms) que tarda el kernel en tramitar un...		3/1.5 - 0/3	15,1 ms
Latencia KernelWrite	Muestra el tiempo (en ms) que tarda el kernel en tramitar un...		3/1.5 - 0/3	22,2 ms

4.6.- Oracle Database Server

4.6.1.- Análisis del Sistema

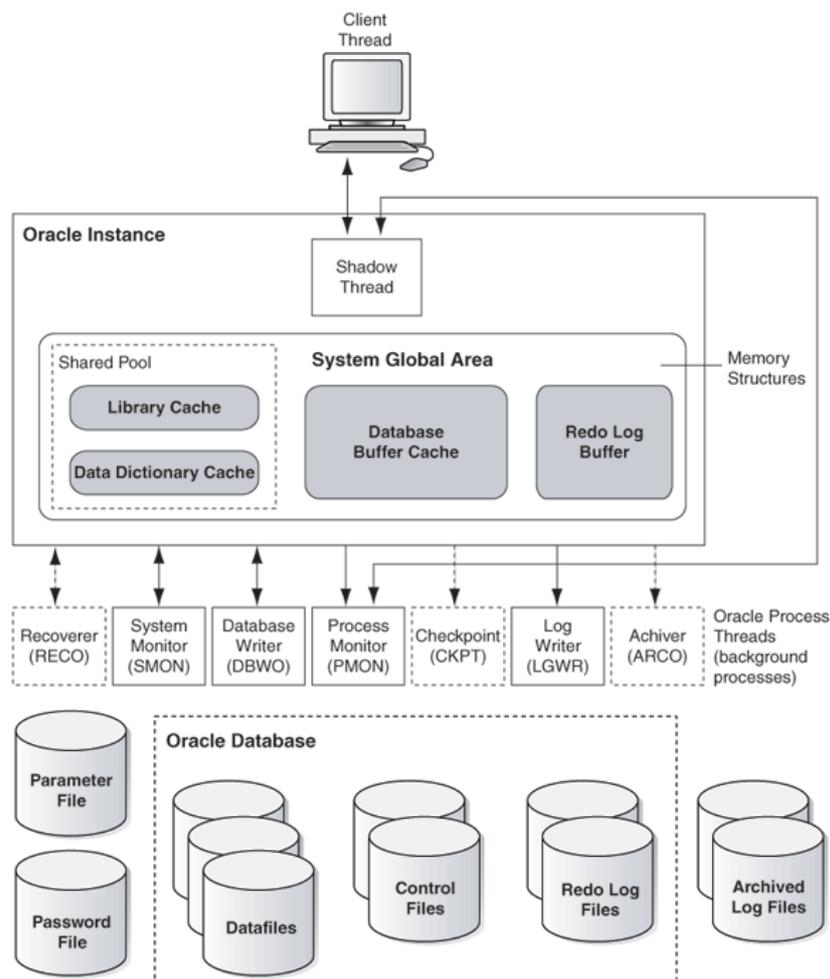
Oracle Database Server, también conocido como Oracle RDBMS (*Relational Data Base Management System*) o simplemente Oracle, es un sistema de gestión de bases de datos de Oracle Corporation actualmente muy extendido en entornos empresariales.



Como la mayoría de herramientas actuales, cuenta con una aplicación de administración accesible vía web (anteriormente mediante una aplicación cliente Java de escritorio) que recibe el nombre de Oracle Enterprise Management Console. Es altamente probable que utilicemos esta herramienta para obtener parámetros interesantes que sean susceptibles de ser monitorizados y para contrastar la información recabada mediante Pandora FMS.

4.6.1.1.- Arquitectura Básica

Un servidor Oracle está compuesto por dos entidades diferenciadas:



- **Instancia:** Compuesta de una estructura de memoria que recibe el nombre de *SGA* o *System Global Area* (almacenada en la RAM) y una serie de procesos de fondo. La *SGA* se compone de varias estructuras de memoria:
 - **Database Buffer Cache:** Área de trabajo SQL en RAM, que actúa en modo caché para minimizar el número de accesos a disco.
 - **Redo Log Buffer:** Ídem, aplicado a los *Redo Logs* (detallados en la estructura física).
 - **Shared Pool, Java Pool, Large Pool, Stream Pool...** : Estructuras más complejas (de caché, generalmente) que no entraremos a analizar.
- **Estructura Física:** Hace referencia a los ficheros físicos almacenados, que son de tres tipos:
 - **Datafiles:** Ficheros correspondientes a los datos de las bases de datos. No se corresponden directamente con las tablas: las tablas pertenecen al nivel lógico y los ficheros de datos con el nivel físico (los procesos de *Oracle* se encargan de relacionarlos).
 - **Redo Logs:** Es un registro secuencial de los cambios aplicados a los datos, que protegen al sistema de pérdidas de datos (permitiendo el *rollback*).
 - **Control Files:** Almacenan información acerca de la ubicación de las estructuras físicas de *Oracle*.

4.6.1.2.- Acceso a Información de Estado y Rendimiento

Oracle almacena información de rendimiento del sistema, información de estado, información de configuración y demás metadatos dentro de tablas y bases de datos especiales, por lo que nuestra herramienta principal para extraer información será *SQL*Plus*. Esta utilidad permite realizar consultas SQL a un servidor *Oracle* remoto. En *RHEL/CentOS*, se utiliza la aplicación de línea de comando "*sqlplus*", sobre la cual construiremos distintos *wrappers* (para *shell*) con el fin de tratar la información y presentarla en *Pandora FMS*.

También es posible instalar y activar agentes *SNMP* sobre *Oracle*. Los ficheros *MIB* están disponibles públicamente en la web, por lo que éste puede ser un método alternativo para extraer información.

Se contemplará, además, la posibilidad de instalar el agente de *Pandora FMS* sobre el sistema operativo sobre el que se ejecuta *Oracle*. Tal y como sucedía con otros dispositivos, por motivos de limpieza, optaremos por evitar esta opción salvo que resulte estrictamente necesaria.

En resumen, las alternativas para extraer información son:

- Cliente *Oracle*, instalándolo en *Pandora Server*.
- *SNMP*, activando los agentes necesarios.
- Agente *Pandora*, instalándolo en el servidor *Red Hat* que aloja *Oracle*.
- *SSH*, estableciendo una conexión con el servidor *Red Hat* y sirviéndonos de las herramientas de cliente *Oracle* instaladas en él.

4.6.2.- Entorno de Prueba

4.6.2.1.- Instancia Oracle

Para realizar las pruebas, se nos otorga acceso a un sistema en preproducción (por lo que no podremos manipularlo de modo que ponga en riesgo su integridad y funcionamiento). Se nos otorgan privilegios de *DBA* (*Data Base Administrator*), si bien, por las razones mencionadas, realizaremos las acciones necesarias con cautela.

Cabe destacar que el servidor *RHEL* que aloja la instancia de *Oracle* se utiliza también para alojar otro tipo de servicios, como un servidor *VPN* (empleado en secciones anteriores) o un servidor de aplicaciones *Tomcat* (que se tratará en el próximo capítulo).

4.6.2.2.- Instalación de Cliente Oracle en Pandora Server

El cliente de *Oracle* resulta notablemente más pesado (~ 600 MB) que otras aplicaciones cliente de gestión instaladas anteriormente. Además, esta instalación requiere de una interfaz gráfica para lanzar el instalador.

Dado que nuestro servidor *CentOS* se creó a partir de una instalación *minimal* (sin interfaz gráfica, a fin de mantener ligera la *VM*), optamos por instalar el *X Window System* y configurar un servidor *X* (mediante la utilidad *Cygwin*)¹⁴ sobre la máquina física *Windows* del puesto de trabajo.

Por tanto, con esta solución logramos que la carga gráfica recaiga sobre la máquina *Windows* en lugar de la *VM Pandora Server*.

4.6.3.- Creación de Agentes

Decidimos crear un agente exclusivo para la instancia de *Oracle* a la que se nos otorga acceso, en lugar de un único agente para el servidor (que, a la postre, podría agrupar una cantidad incómoda de módulos asociados a distintos servicios).

El agente creado es:

- **Oracle Kanpoko Txinzer**

¹⁴ Por evitar sobrecargar la documentación, al igual que en otros casos, obviamos el proceso de instalación y configuración de todas las utilidades *X.Org* y *Cygwin/X Server* en sus respectivas máquinas. Aún así, cabe destacar que este proceso no es trivial y ha supuesto un coste temporal no despreciable.

4.6.4.- Desarrollo de Módulos

4.6.4.1.- Módulos SQL*Plus

Para realizar consultas mediante *SQL*Plus*, generamos un *script* de nombre "*oracle_sqlplus.sh*" que en función de un parámetro (indicado a nivel de *plug-in*) realizará el chequeo correspondientes a su módulo correspondiente¹⁵:

- **Errores ORA:** Muestra todos los errores del tipo *ORA-XXXXX* (siendo *XXXXX* un identificador numérico del error) presentes en el sistema. Solo se mostrará *OK* cuando el sistema esté libre de errores. En caso de existir un error se mostrará un estado *critical*.
- **Sesiones Activas:** Muestra la relación entre sesiones activas y el número máximo de sesiones permitidas en *Oracle*. Se muestra *warning* entre 75-90% y *critical* para valores superiores.
- **Procesos Activos:** Muestra la relación entre el número de procesos de *Oracle* activos en el sistema y el máximo establecido. Mostramos *warning* entre 75-90% y *critical* para valores superiores.
- **Transacciones en Curso:** Muestra la relación entre las transacciones en curso y el número máximo de transacciones permitidas en *Oracle*. Se muestra *warning* entre 75-90% y *critical* para valores superiores.

4.6.4.2.- Módulos Adaptados de Nagios Core

Estos módulos se han creado adaptando a nuestras necesidades el *script* en lenguaje *Perl* "*check_oracle_instant*" obtenido de *Nagios Exchange* (repositorio oficial de *plug-ins* desarrollados por la comunidad *opensource* para la plataforma *Nagios*).

Dicho *script*, renombrado a "*oracle_db_estado.pl*", muestra por pantalla una serie de parámetros relativos al rendimiento de la base de datos. Esta nueva versión nos permite generar módulos a partir del mismo *script* ejecutado en distintos modos (distintos parámetros indicados a nivel de *plug-in*). Cabe destacar que, en la descripción de los módulos, se ha añadido un pequeño texto que ayude a comprender el por qué de los valores anormales, y en su caso, una posible solución.

Los módulos desarrollados son:

- **Dictionary Cache Hit Ratio:** Es el ratio de acierto (%) de caché para consultas sobre el diccionario. El diccionario es un conjunto de tablas de solo lectura que contienen información relacionada con la base de datos.

¹⁵ Nótese que el clásico *Host Alive* no tiene cabida en esta lista de módulos, ya que según el esquema que hemos decidido seguir dicho módulo correspondería al servidor o sistema operativo como tal y no al servidor de base de datos.

El valor mínimo recomendado es de 90%, ya que, según *Oracle*, un valor por debajo de dicha cifra indica un mal rendimiento del sistema (causado, por ejemplo, por escasez de memoria reservada para el *Shared Pool*). En *Pandora*, será *warning* entre 90-92% y *critical* para valores inferiores a 90%.

También se indica que esta cifra suele ser inferior, transitoriamente, en el periodo desde que se inicia *Oracle* hasta que la caché va normalizándose de forma progresiva (las primeras consultas, cuando la caché esté vacía, siempre supondrán un fallo de caché).

- **Library Cache Hit Ratio:** Es el ratio de acierto de caché para sentencias *SQL*. Análogamente, *Oracle* indica que un ratio de fallos superior al 1% representa escasez de memoria reservada para el *Shared Pool*. En *Pandora*, será *warning* entre 99-99.2% y *critical* para valores inferiores a 99%.
- **DB Block Buffer Cache Hit Ratio:** Es el ratio con el que *Oracle* encuentra en memoria los bloques de datos que necesita la base de datos objetivo. Un valor bajo de este valor representa un exceso de accesos a disco.

Este valor no es un indicador directo del: es posible que un valor más bajo de lugar a un mejor rendimiento general del sistema. Aún y todo, un valor inferior al 85% supondrá, en la inmensa mayoría de casos, un lastre para el rendimiento de la base de datos. En *Pandora*, será *warning* entre 85-90% y *critical* para valores inferiores a 85%.

- **Latch Hit Ratio:** Indica el rendimiento de los mecanismos de cerrojo (*lock/unlock*). Un valor bajo indica problemas relacionados con estos mecanismos. Varias fuentes difieren en el mínimo concreto que debería poseer, si bien todas indican que el valor siempre debe ser muy cercano al 100%. Por ello, establecemos que el módulo será *warning* entre 98-99% y *critical* para valores inferiores a 98%.
- **Disk Sort Ratio:** Es el ratio de cláusulas "*ORDER BY*" que han sido aplicadas sobre una cantidad de datos demasiado grande, rebasando la memoria disponible y provocando el acceso a disco para completar las operaciones. Un valor superior al 5% indica la necesidad de incrementar la reserva de memoria en determinadas estructuras de memoria (como el *Large Pool* o el valor *sort_area_size*). En *Pandora*, será *warning* entre 4-5% y *critical* para valores superiores al 5%.
- **Rollback Segment Wait Ratio:** Un segmento *rollback* es un objeto que contiene versiones previas de los datos escritos en la base de datos. Este ratio indica la cantidad de veces que se ha tenido que esperar para acceder a un segmento *rollback* en relación con el número total de accesos a segmentos *rollback*.

Una espera surge cuando varias sesiones compiten por acceder a un segmento *rollback* o cuando el canal de entrada/salida al disco está ocupado. Un valor superior al 5% se considera problemático, por lo que en *Pandora* será *warning* entre 4-5% y *critical* para valores superiores al 5%.

- **Dispatcher Workload:** Los *dispatchers* de Oracle son procesos encargados de dirigir las peticiones de inicio de sesión externas a los procesos internos del sistema. Este valor indica la carga de trabajo media de los *dispatchers*, que crecerá junto con el número de peticiones de apertura de sesión a través de la red.

Oracle recomienda mantener este valor por debajo del 50% para mantener el rendimiento y la estabilidad del sistema. Si el *workload* medio rebasa este valor, se recomienda aumentar el número de *dispatchers* para distribuir la carga de trabajo. *Pandora* mostrará *warning* entre el 45-50% de carga y *critical* para valores superiores al 50%.

4.6.4.3.- Módulos Oracle NET

El último método de extracción de información emplea la utilidad *TNSPING* (clasificada en el "*Oracle NET Bundle*", dentro de las herramientas cliente de Oracle). El script en el que se basa el *plug-in* es "*oracle_listener.sh*" siendo el único módulo el siguiente:

- **Listener Estado:** Comprueba que el *listener* de Oracle está en ejecución en un puerto determinado y responde correctamente. De no ser así, el módulo muestra estado *critical*.

4.6.5.- Configuración de Alertas

Como en casos anteriores, no se han configurado alertas que pudieran resultar de interés.

4.6.6.- Evaluación de Soluciones

4.6.6.1.- Oracle Enterprise Manager Console

Nuestra referencia para contrastar la información de los módulos será la herramienta de gestión de Oracle: la *Enterprise Manager Console*. Al igual que muchas otras herramientas de gestión, se accede mediante *HTTPS* a una aplicación web (en este caso basada en Java, como era de esperar).

Por la configuración del *firewall*, no nos es posible acceder directamente al puerto en el que está a la escucha el servicio web (por defecto el *TCP/1158*). Por ello, para evitar que la configuración del sistema se vea afectada, establecemos una sesión *SSH* redirigiendo el puerto local *TCP/1158* al puerto remoto *TCP/1158* de la interfaz interna (no protegida por el *firewall*).

El acceso, por tanto, se realizará desde el navegador web indicando la siguiente *URL*:

`https://127.0.0.1:1158/em`

El aspecto de la interfaz es el siguiente:

The screenshot displays the Oracle Enterprise Manager 11g Database Control interface for the database instance 'oradata.dstu.net'. The page is organized into several sections:

- General:** Shows the instance status as 'Up', with 'Up Since' on Oct 30, 2007 5:33:17 PM EDT. Other details include Instance Name 'oradata', Version '11.1.0.6.0', Host 'ServerRHEL5.dstu.net', Listener 'LISTENER_ServerRHEL5.dstu.net', and ASM '+ASM_ServerRHEL5.dstu.net'. There are 'Shutdown' and 'Black Out' buttons.
- Host CPU:** A bar chart showing CPU usage for 'Other' and 'oradata'. The load is 0.49 and paging is 0.00. Maximum CPU is 2.
- Active Sessions:** A bar chart showing session types: Wait, User I/O, and CPU. Maximum CPU is 2.
- SQL Response Time:** A bar chart comparing 'Collection' and 'Reference' times. The SQL Response Time (%) is 1445.54.
- Diagnostic Summary:** Shows 'No ADDM run available', 'No ORA- errors' in the alert log, and 0 active incidents.
- Space Summary:** Lists Database Size (1.492 GB), Problem Tablespaces (0), Segment Advisor Recommendations (0), and Policy Violations (0).
- High Availability:** Shows Instance Recovery Time (18 sec), Last Backup (n/a), Usable Flash Recovery Area (92.04%), and Flashback Database Logging (Disabled).

4.7.- Servidores de Aplicaciones Web *Tomcat*

4.7.1.- Análisis del Dispositivo

Apache Tomcat es un servidor de aplicaciones web que soporta *servlets* y *JSPs* (*Java Server Pages*), desarrollado en el *Jakarta Project* de la *Apache Software Foundation*.



Tomcat puede funcionar como servidor web por sí mismo, si bien es habitual que su motor de *servlets* se encuentre en combinación con un servidor web *Apache*.

Al estar escrito en *Java*, *Tomcat* es soportado por cualquier sistema operativo que disponga de la máquina virtual *Java*.

4.7.1.1.- Ficheros de Configuración

Los ficheros de configuración de *Tomcat* se almacenan, generalmente, bajo la ruta `"/etc/tomcat5"` o sus análogas para las distintas versiones del servidor (que actualmente serían la 5, 6 o 7). Los ficheros de mayor relevancia son los siguientes:

- ***tomcat.conf***: Contiene las rutas de instalación de librerías y componentes de *Tomcat*, como *Catalina* (contenedor de *servlets*) o *Jasper* (compilador de ficheros *JSP*, para convertirlos en *servlets* que *Catalina* pueda manipular).
- ***server.xml***: Fichero de configuración principal que define qué puertos están a la escucha de peticiones *HTTP(S)* y qué servicio las atenderá. Se indican, adicionalmente, valores máximos de espera (*timeout*) y número máximo de *threads* que pueden generarse.
- ***tomcat-users.xml***: Almacena la relación de usuarios y sus respectivos roles dentro de *Tomcat*, además de contraseñas en texto claro y sin

cifrar¹⁶. Estas credenciales pueden requerirse para acceder a determinados servicios, como acceder al panel de administración que mostraremos en los siguientes subapartados.

4.7.1.2.- Extracción de Información

Para extraer información de rendimiento y estado de servidores *Tomcat* desde *Pandora FMS*, podríamos recurrir a las siguientes técnicas:

- Descargando una página *HTML* generada al invocar mediante *HTTP* a la aplicación web que extrae y presenta la información deseada.
- Ejecutando desde la línea de comandos la aplicación web mencionada. Requiere una conexión mediante *SSH* al servidor remoto o la instalación de un agente de *Pandora FMS* en el mismo.
- Desarrollando una serie de aplicaciones dentro del servidor remoto que extraigan la información requerida. Estas aplicaciones deberían programarse en lenguaje *Java* y emplear la *API Java Management Extensions (JMX)*. Del mismo modo, resultaría necesario establecer una conexión *SSH* o instalar un agente *Pandora* en el servidor para poder ejecutar las aplicaciones desarrolladas.

La solución escogida se detalla en el apartado de desarrollo de módulos.

4.7.2.- Entorno de Prueba

Se tratará con un servidor *Tomcat* versión 5 en un entorno de preproducción, que comparte máquina con el servidor de bases de datos *Oracle* monitorizado en el capítulo anterior.

Adicionalmente, decidimos instalar nuestro propio servidor *Tomcat v6* en la máquina virtual *CentOS* sobre la que se ejecuta *Pandora FMS*¹⁷. Esto nos permitirá manipular la configuración y forzar que surjan situaciones anómalas sin afectar a los sistemas de la empresa.

4.7.3.- Creación de Agentes

Asociados a nuestro entorno de prueba, creamos los agentes:

- ***Tomcat Kanpoko Txinzer***
- ***Tomcat Pandoraserver***

¹⁶ Comprobado únicamente en las versiones 5 y 6 de *Tomcat*. Incluso en estas versiones, es posible realizar este proceso de autenticación de un modo más seguro mediante servidores *RADIUS*, *LDAP* o *AD*.

¹⁷ La instalación de *Tomcat* mediante el gestor de paquetes *yum* resulta trivial.

4.7.4.- Desarrollo de Módulos

4.7.4.1.- Módulos con Fichero Caché

Estos módulos se han generado a partir del *script* "*tomcat_info.sh*", que descarga el fichero *XML* generado por la aplicación web que atiende a la *URL*:

```
http://servidor:puerto/manager/status?XML=true
```

Para minimizar el tráfico de red, este fichero *XML* se descargará periódicamente, de modo que cada módulo lo consulte desde el sistema de ficheros local.

Asimismo, para evitar tener que crear una tarea *cron*, cada ejecución del *script* comprobará que el fichero *XML* temporal ha sido descargado hace menos de 5 minutos. De no ser así (o de no existir), el fichero será descargado antes de ser procesado por el propio *script*.

Los módulos desarrollados de esta forma son los siguientes:

- **Memoria Uso:** Uso de memoria por parte de la *JVM* que soporta a *Tomcat*. Se muestra la relación entre la memoria total reservada y la memoria máxima que puede ocupar. Mostramos *warning* entre 75-90% y *critical* para valores superiores.
- **Threads:** Muestra la relación entre el número actual de *threads* lanzados por *Tomcat* y el máximo permitido (según se establece en *server.xml*). Mostramos *warning* entre 75-90% y *critical* para valores superiores.
- **Peticiones/s:** Peticiones *HTTP* o *HTTPS* totales recibidas por segundo.
- **Errores/s:** Número de peticiones erróneas (por ejemplo, peticiones de páginas o ficheros no existentes) recibidas por segundo.
- **Tráfico Bajada:** Bytes por segundo que descarga *Tomcat*. Raramente se encuentra un tráfico de bajada significativo, salvo en aquellos casos en los que se permita la subida de ficheros al servidor.
- **Tráfico Subida:** Bytes por segundo transmitidos por *Tomcat*.
- **Tiempo Procesamiento:** Porcentaje de tiempo que *Tomcat* ha estado procesando peticiones de los distintos clientes (desde que se ha recibido la petición hasta que ésta ha sido respondida). Éste valor crece cuando se realizan peticiones a servicios pesados, que realizan múltiples operaciones antes de devolver un resultado.

***NOTA:** Este módulo es experimental y, a priori, no supone un indicador fiable del rendimiento de *Tomcat*. No se establecen límites

para definir los estados *warning* o *critical*. El objetivo de este módulo es, por el momento, el de satisfacer la curiosidad y observar su comportamiento cuando el número de *threads* utilizados por *Tomcat* es elevado.

4.7.4.2.- Módulos sin Fichero Caché

Este módulo se ha desarrollado con el *script* "*tomcat_app.sh*", que consulta la *URL*:

```
http://servidor:puerto/manager/list
```

Devolviendo una línea de texto por cada servicio activo en *Tomcat*. Dado que el tamaño del fichero de texto es mínimo, no realizaremos una copia-caché como en el caso anterior sino que descargaremos directamente el fichero en cada ejecución del *script*.

El módulo es:

- **App:** Muestra si la aplicación indicada por parámetro está lanzada. De no ser así, el módulo mostrará *critical*.

4.7.5.- Configuración de Alertas

De nuevo, no se han configurado alertas que pudieran resultar de interés.

4.7.6.- Evaluación de Soluciones

Al igual que en casos anteriores, dado que disponemos de un servidor *Tomcat* que no forma parte de los sistemas en producción de la empresa, tendremos una libertad adicional para evaluar los módulos:

4.7.6.1.- Peticiones Masivas de Descarga de Páginas Estáticas

En este apartado, programaremos la descarga automática y masiva de páginas disponibles en el servidor *Tomcat*, para posteriormente observar si esta situación de carga elevada queda reflejada en los módulos que hemos creado.

Observamos el estado de los módulos antes de realizar la prueba:

Memoria Uso	Uso de memoria por parte de la JVM (total / max)		90/75 - 0/90	3,1 %
Threads	Uso de threads por parte de Tomcat (threads actuales / máxim...		90/75 - 0/90	1,5 %
Networking				
Errores/s	Peticiones erróneas recibidas por segundo		N/A - N/A	0 errores/s
Peticiones/s	Peticiones HTTP/HTTPS recibidas por segundo		N/A - N/A	0 peticiones/s
Tráfico Bajada	Bytes/s de bajada de Tomcat		N/A - N/A	0 bytes/s
Tráfico Subida	Bytes/s de subida de Tomcat		N/A - N/A	0 bytes/s
Application				
App sample	Muestra el estado del servicio web "sample"		N/A - N/A	  101
Performance				
Tiempo Procesamiento	Porcentaje de tiempo que Tomcat ha estado procesando peticio...		N/A - N/A	0 %

Desde la propia *VM pandoraserver* (evitando que el tráfico de red atraviese los *switches* de la empresa), lanzamos los siguientes comandos:

```
while true; do wget -q -O /dev/null http://127.0.0.1:8080/;
wget -q -O /dev/null http://127.0.0.1:8080/error; echo -n ".";
done
```

El objetivo de este bucle es descargar sin pausa una página concreta de *Tomcat (/index.html)* y hacer peticiones de una página inexistente (*/error*). En cada ciclo se imprime un punto por pantalla:

```
[root@pandoraserver ~]# while true; do wget -q -O /dev/null http://127.0.0.1:8080/;
wget -q -O /dev/null http://127.0.0.1:8080/error; echo -n "."; done
.....
.....
.....
```

El resultado es el siguiente:

Memoria Uso	Uso de memoria por parte de la JVM (total / max)		90/75 - 0/90	3.1 %
Threads	Uso de threads por parte de Tomcat (threads actuales / máxim...		90/75 - 0/90	1.5 %
Networking				
Errores/s	Peticiones erróneas recibidas por segundo		N/A - N/A	46.0 <i>errores/s</i>
Peticiones/s	Peticiones HTTP/HTTPS recibidas por segundo		N/A - N/A	92.0 <i>peticiones/s</i>
Tráfico Bajada	Bytes/s de bajada de Tomcat		N/A - N/A	0 <i>bytes/s</i>
Tráfico Subida	Bytes/s de subida de Tomcat		N/A - N/A	399,799.5 <i>bytes/s</i>
Application				
App sample	Muestra el estado del servicio web "sample"		N/A - N/A	
Performance				
Tiempo Procesamiento	Porcentaje de tiempo que Tomcat ha estado procesando peticio...		N/A - N/A	1.8 %

Observamos que los 400 *KB/s* de tráfico generado están lejos de causar una situación de sobrecarga en *Tomcat*, aunque sí logramos apreciar una mayor actividad del sistema.

Al tratarse de peticiones secuenciales (no se lanza una petición hasta haber finalizado la anterior), el uso de hilos se ha mantenido estable. El hecho de realizar peticiones de descarga de páginas en lugar de invocar aplicaciones complejas ha favorecido que el tiempo de procesamiento y el uso de memoria de la *JVM* apenas hayan sufrido cambios.

4.7.6.2.- Peticiones Masivas de Ejecución de Aplicaciones

Dado que la evaluación anterior no consiguió afectar a los valores de uso de *threads* y memoria, nos disponemos a realizar un test de estrés mucho más agresivo que el anterior.

Lanzaremos múltiples bucles que realizarán peticiones a aplicaciones que requieran cierto procesamiento, tanto desde el propio *pandoraserver* como desde la máquina *Windows* local. Para evitar sobrecargar la red de la empresa, creamos una ruta estática desde la máquina *Windows* para

alcanzar la *VM pandoraserver* directamente, sin atravesar ningún otro dispositivo de red. Lo comprobamos mediante el comando "*tracert*".

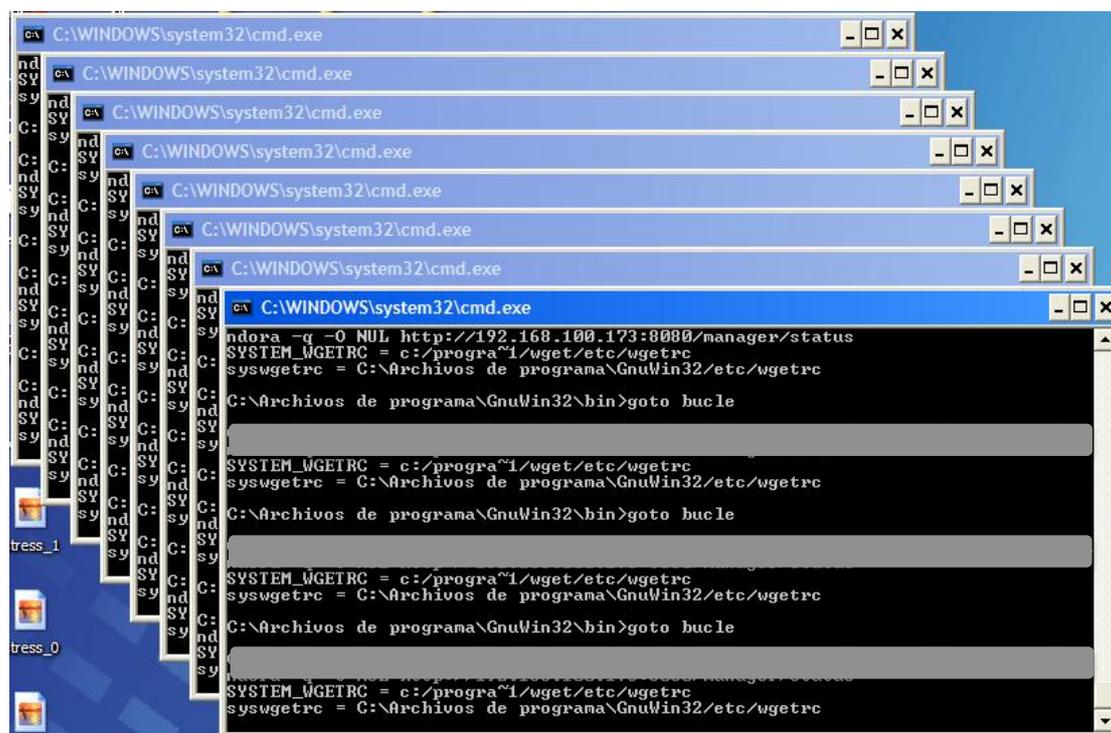
En *Windows*, creamos un fichero *batch* (*TomcatStress.bat*) simple:

```
:bucle

wget --http-user=XXX --http-password=XXX -q -O NUL
http://XXX.XXX.XXX.XXX:8080/manager/status

goto bucle
```

Ejecutamos hasta 8 veces el pequeño *script* creado, de manera simultánea:



En la *VM*, creamos otro pequeño *script* (*TomcatStress.bash*):

```
while true; do wget -q -O
/dev/null --http-user=XXX --http-password=XXX
http://127.0.0.1:8080/manager/status/all; echo -n "."; done
```

Y lo ejecutamos 4 veces simultáneas, como tarea de fondo:

```
# ./TomcatStress.sh&
```

Pese a nuestros esfuerzos, observamos que las páginas disponibles en el servidor *Tomcat* se cargan desde *Internet Explorer* sin ninguna demora aparente.

Transcurridos 10 minutos, cerramos todas las ventanas *CMD* en *Windows* y todos los procesos de fondo lanzados en *Linux*. Observamos los resultados en *Pandora FMS*:

Memoria Uso	Uso de memoria por parte de la JVM (total / max)		90/75 - 0/90	22.3 %
Threads	Uso de threads por parte de Tomcat (threads actuales / máxim...		90/75 - 0/90	10 %
Tiempo Procesamiento	Porcentaje de tiempo que Tomcat ha estado procesando peticio...		N/A - N/A	239.1 %

Los módulos parecen reflejar con bastante fidelidad el estado del servidor *Tomcat*: existe una carga de trabajo notable, aunque no llega a lastrar significativamente al servidor de aplicaciones.

Gracias al paralelismo a nivel de *thread*, observamos que el uso del tiempo en procesamiento ha superado el 100% durante el periodo de estrés de *Tomcat*.

4.7.6.3.- Parada de un Servicio Web

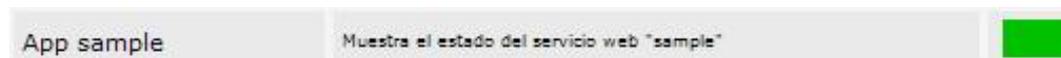
Desde la interfaz web de administración de *Tomcat*, accesible en:

<http://servidor:8080/manager/html/list>

Detenemos la aplicación */sample* que monitorizamos en *Pandora FMS* y observamos el *output* del módulo:



De nuevo, acudimos a la interfaz web de administración para devolver el servicio web a su estado normal:



Referencias/Enlaces del Capítulo

***NOTA:** Los enlaces han sido comprobados por última vez el día 05/06/2013. Es posible que algunos de ellos estén rotos o hayan sido actualizados. En tal caso, conviene acceder al directorio raíz o página inicial e iniciar la búsqueda de la referencia.

- **Equipamento WiMax Alvarion:**
 - *Alvarion BreezeNET Portfolio:*
<http://www.alvarion.com/es/products/product-portfolio/breezenet/breezenet-b>
 - *Alvarion BreezeNET Ralco Networks:*
<http://www.ralco-networks.com/descargas/AlvarionBnetB.pdf>
 - *Alvarion BreezeULTRA Portfolio:*
<http://www.alvarion.com/products/product-portfolio/breezeultra>
 - *Alvarion BreezeULTRA Winncom:*
http://www.winncom.com/pdf/Alvarion_BreezeULTRA_P6000_350/Alvarion_BR_BreezeULTRA_reva_LR.pdf
 - *Alvarion BreezeULTRA Ordering Information:*
http://www.alvarion.com/addons/mailers/announcements/partners/ProductLaunchSite/BreezeULTRA/SalesToolKit-ULTRA/BreezeULTRA_Ordering_Information_v2.pdf
 - *Modelos Alvarion BreezeULTRA:*
http://www.alvarion.com/addons/mailers/announcements/partners/ProductLaunchSite/BreezeULTRA/SalesToolKit-ULTRA/PPT_BreezeULTRA_Product_Presentation_External1.pdf
 - *MIB-2:*
<http://www.oidview.com/mibs/0/RFC1213-MIB.html>
 - *BRIDGE-MIB:*
<http://www.oidview.com/mibs/0/BRIDGE-MIB.html>
 - *Private BreezeACCESS VL MIB:*
<http://www.oidview.com/mibs/12394/ALVARION-DOT11-WLAN-MIB.html>
 - Manual *BreezeULTRA* suministrado directamente por *Alvarion:*
[BreezeULTRA_System_Manual.pdf](#)
 - Ficheros *MIB* suministrados directamente por *Alvarion:*
[alvarion_top.mib](#)
[BMAX-Extreme-TC.mib](#)
[BMAX-Extreme-CM.mib](#)
[BMAX-Extreme-FM.mib](#)
[BMAX-Extreme-PM-mib](#)
[brzAccVL.mib](#)
[vIng.mib](#)
- **Equipamento Networking Industrial Hirschmann:**
 - *MACH102-8TP-R Shopping:*
<http://www.industrialnetworking.com/Manufacturers/Hirschmann-Fast-Ethernet-MACH-100-Switches/Hirschmann-MACH102-8TP-R-Switch>

- *MACH102-8TP-R* Especificaciones *PDF*:
<http://www.industrialnetworking.com/pdf/943969101.pdf>
- *MACH102-8TP-R* Descargas:
<http://www.e-catalog.beldensolutions.com/link/57078-24455-49848-85612-40981/en/conf/0;jsessionid=74256637D9FAF150173996D634202C27>
- *MACH1000 Shopping*:
<http://www.industrialnetworking.com/Manufacturers/Hirschmann-MACH1040-Ethernet-Switches/Hirschmann-MACH1000-Switch-MAR1042-4C4C4C4C9999SLLHR>
- *MACH1000* Especificaciones *PDF*:
<http://www.industrialnetworking.com/pdf/MAR1042-4C4C4C4C9999SLLHR.pdf>
- *MACH1000* Descargas:
<http://www.e-catalog.beldensolutions.com/link/57078-24455-49846-49997-82828-173144-173176/en/MAR1042-4C4C4C4C9999SLMHPHH07.0.0>
- Ficheros *MIB* suministrados por *Belden*:
hmARC.mib
hmdhcps.mib
hmDVMRP.mib
HMIOMODULE-SNMP-MIB.mib
hmLLDP.mib
hmPIM.mib
hmPlatform4.mib
hmPlatform4Mcast.mib
hmPlatform4Qos.mib
hmPlatform4Routing.mib
hmpriv.mib
hmRingRed.mib
hmtracking.mib
lldp.mib
lldp_dot1.mib
lldp_dot3.mib
lldp_hm.mib
lldp_med.mib
lldp_pno.mib
usrgrp.mib
- *Plug-In* para medir la pérdida de paquetes:
http://pandorafms.com/index.php?sec=Library&sec2=repository&lng=es&action=view_PUI&id_PUI=227
- **Servidores *HP ProLiant*:**
 - *HP ProLiant Servers*:
<http://h18000.www1.hp.com/products/servers/platforms/index.html>
 - *HP iLO*:
<http://www.hp.com/servers/lights-out>
 - *HP iLO: ProActive Insight Architecture*:
<http://www.youtube.com/watch?v=EkCiGKKnreA>

- *HP ProLiant DL360 Specs PDF:*
http://h18000.www1.hp.com/products/quickspecs/13598_na/13598_na.pdf
- *HP ProLiant DL360 G7 Overview:*
<http://h10010.www1.hp.com/wwpc/us/en/sm/WF05a/15351-15351-3328412-241644-241475-4091408.html?dnr=1>
- *HP ProLiant DL380 Specs PDF:*
http://h18000.www1.hp.com/products/quickspecs/13595_na/13595_na.pdf
- *HP ProLiant DL380 Overview:*
<http://h10010.www1.hp.com/wwpc/us/en/sm/WF05a/15351-15351-3328412-241644-241475-4091412.html?dnr=1>
- *HP iLO Subsystem:*
<http://h20584.www2.hp.com/hpgt/guides/select?lang=es&cc=pr&prodTypeId=329290&prodSeriesId=4154735>
- *HP iLO3 User Guide PDF:*
<http://bizsupport2.austin.hp.com/bc/docs/support/SupportManual/c02774507/c02774507.pdf>
- *HP iLO Security PDF:*
<http://bizsupport2.austin.hp.com/bc/docs/support/SupportManual/c00212796/c00212796.pdf>
- *Intel IPMI-over-LAN 2.0:*
<http://www.intel.com/content/www/us/en/servers/ipmi/ipmi-home.html>
- *"The Power of IPMI" by TK Servers, PDF:*
http://www.netways.de/uploads/media/Werner_Fischer_The-Power-Of-IPMI.pdf
- *Open IPMI Tools for Linux:*
<http://ipmitool.sourceforge.net/index.html>
- *IPMITool Manpage Sourceforge:*
<http://ipmitool.sourceforge.net/manpage.html>
- *Management of VMware ESXi on HP ProLiant servers PDF:*
<http://www.vmware.com/files/pdf/VMware-ESXi-HPProLiantServers-WP-EN.pdf>
- *6 Simple Steps: Monitoring ESXi with HP SIM:*
<http://h30507.www3.hp.com/t5/Technical-Support-Services-Blog/6-Simple-Steps-to-Monitoring-ESXi-with-Insight-Remote-Support/ba-p/100789>
- *HP VMware ESXi management environment PDF:*
<http://h20000.www2.hp.com/bc/docs/support/SupportManual/c02000740/c02000740.pdf>
- *vSphere Management Assistant (vMA):*
<http://www.vmware.com/go/vima>
- *vMA Guide PDF:*
http://pubs.vmware.com/vsphere-51/topic/com.vmware.ICbase/PDF/vma_51_guide.pdf
- *HP iLO2 Firmware Update:*
<http://h20000.www2.hp.com/bizsupport/TechSupport/SoftwareDescription.jsp?lang=en&cc=us&prodTypeId=15351&prodSeriesId=1146658&prodNameId=1135772&swEnvOID=1005&swLang=>

[8&mode=2&taskId=135&swItem=MTX-ab90093287a14896b02f537b7b](http://my.vmware.com/web/vmware/details?productId=285&downloadGroup=HP-ESXI-5.1.0-GA-10SEP2012)

- *HP Custom Image for VMware ESXi 5.1:*
<https://my.vmware.com/web/vmware/details?productId=285&downloadGroup=HP-ESXI-5.1.0-GA-10SEP2012>
 - *HP Custom Image for VMware ESXi 5.1 Release Notes PDF:*
<http://h10032.www1.hp.com/ctg/Manual/c03469594.pdf>
 - *VMware vSphere Client 5.1 download:*
https://my.vmware.com/web/vmware/info/slug/datacenter_cloud_infrastructure/vmware_vsphere/5_1
 - *VMware vSphere CLI 5.1 download:*
<https://my.vmware.com/web/vmware/details?downloadGroup=VSP510-VCLI-510&productId=285>
 - *HP Insight Management WBEM Providers for ESX PDF:*
<http://bizsupport2.austin.hp.com/bc/docs/support/SupportManual/c03537421/c03537421.pdf>
 - *WBEM Scripting Tools for ESXi - PyWBEM HowTo PDF:*
ftp://ftp.compaq.com/pub/products/servers/vmware/esx3i/HP_WBEM_Provider_for_ESXi_Scripting_Tools_Guide.pdf
 - *PyWBEM Examples:*
http://pywbem.sourceforge.net/apps/mediawiki/pywbem/index.php?title=Client_Examples
 - *WBEM Providers for HP Integrated VMware vSphere 5.1 PDF:*
<http://h10032.www1.hp.com/ctg/Manual/c03478329.pdf>
- **Sistemas de Almacenamiento EMC²:**
 - *VNX Series:*
<http://spain.emc.com/storage/vnx/vnx-series.htm>
 - *VNX Series Datasheet PDF:*
<http://spain.emc.com/collateral/hardware/data-sheets/h8520-vnx-family-ds.pdf>
 - *VNXe Series:*
<http://spain.emc.com/storage/vnx/vnx-series.htm>
 - *VNXe Series Datasheet PDF:*
<http://spain.emc.com/collateral/hardware/data-sheets/h8516-vnx-e-series-ds.pdf>
 - *EMC VNX and VNXe Online Store:*
<https://store.emc.com/Product-Family/VNX-AND-VNXe-PRODUCTS/c/VNX?q=%3Arelevance%3AProductSeries%3AVNX>
 - *EMC Unisphere PDF:*
<http://www.emc.com/collateral/software/white-papers/h8017-unisphere-element-manager.pdf>
 - *Área de Descargas Unisphere:*
https://support.emc.com/downloads/8864_VNXe-Series
 - *VNXe Unisphere CLI User Guide PDF:*
https://support.emc.com/docu31491_VNXe-Unisphere-CLI-User-Guide.pdf

- **Entorno de Virtualización VMware:**
 - *VMware ESX vs ESXi:*
<http://www.vmware.com/products/vsphere/esxi-and-esx/compare.html>
 - *vSphere vMotion:*
<http://www.vmware.com/products/datacenter-virtualization/vsphere/vmotion.html>
 - *VMware MIB Download:*
<https://my.vmware.com/web/vmware/details?downloadGroup=VSP510-SNMPMIBS-510&productId=285>
 - *Check VMFS Nagios Exchange:*
http://exchange.nagios.org/directory/Plugins/Operating-Systems/%2A-Virtual-Environments/VMWare/check_vmfs-2Esh--2D-Check-vmfs-datastores-through-VMware-VCLI/details
 - *Using the ESXTOP Utility:*
http://pubs.vmware.com/vsphere-4-esx-vcenter/index.jsp?topic=/com.vmware.vsphere.resourcemanagement.doc_41/performance_monitoring_utilities_resxtp_and_esxtp/c_using_the_resxtp_utility.html
 - *Stress Tool CentOS 6:*
<http://pkgs.org/centos-6-rhel-6/repoforge-i386/stress-1.0.2-1.el6.rf.i686.rpm.html>

 - ***NOTA:** En este capítulo, también se han empleado algunas de las referencias descritas en el capítulo "*Servidores HP ProLiant*"

- **Oracle Database Server:**
 - *Oracle Database Server Overview:*
<http://www.oracle.com/us/products/database/overview/index.html>
 - *Oracle 11g R2 Database Documentation Library:*
http://www.oracle.com/pls/db112/portal.portal_db
 - *Monitoring and Tuning the Database:*
http://docs.oracle.com/cd/E11882_01/server.112/e10897/montune.htm#CHDDAFBD
 - *Monitoring Errors and Alerts:*
http://docs.oracle.com/cd/E11882_01/server.112/e10595/monitoring001.htm
 - Ejemplos de Parámetros de Monitorización de Oracle:
<http://www.manageengine.com/it360/help/meitms/applications/help/monitors/oracle-database-servers.html>
 - *Instant Client Downloads for Linux x86:*
<http://www.oracle.com/technetwork/topics/linuxsoft-082809.html>
 - *Oracle System Tables:*
http://docs.oracle.com/cd/E11882_01/timesten.112/e21644/systemtables.htm
 - *Database Server Monitoring with Nagios:*
<http://nagios.frank4dd.com/howto/db-monitoring.htm>
 - *Nagios Oracle Plugin Database:*
<http://exchange.nagios.org/directory/Plugins/Databases/Oracle>

- *Oracle MIBS:*
<http://www.oidview.com/mibs/111/md-111-1.html>
- *Nagios Plug-In "check_oracle_instant":*
<http://exchange.nagios.org/directory/Plugins/Databases/Oracle/Check-Oracle-status-%26-health-without-install-Oracle-client/details>
- *Oracle Dictionary Cache Hit Ratio:*
http://docs.oracle.com/cd/B16240_01/doc/doc.102/e16282/oracle_database_help/oracle_database_instance_efficiency_dictionaryhit_pct.html
- *Oracle Library Cache Hit Ratio:*
http://docs.oracle.com/cd/B16240_01/doc/doc.102/e16282/oracle_database_help/oracle_database_instance_efficiency_libcache_hit_pct.html
- *Oracle Buffer Cache Hit Ratio:*
http://docs.oracle.com/cd/B16240_01/doc/doc.102/e16282/oracle_database_help/oracle_database_instance_efficiency_bufcachehit_pct.html
- *Oracle-DBA Latch Hit Ratio:*
http://www.dba-oracle.com/m_latch_hit_ratio.htm
- *Oracle In-Memory Sort Ratio:*
http://www.dba-oracle.com/m_in_memory_sort_ratio.htm
- *OracleFAQ Rollback Wiki:*
http://www.oraFAQ.com/wiki/Rollback_Segment
- *Akadia Rollback Segment Tuning:*
http://www.akadia.com/services/scripts/rollback_segment_tuning.sql
- *Oracle Configuring Multi-Threaded Servers:*
http://docs.oracle.com/cd/A87860_01/doc/network.817/a76933/mts.htm
- *Oracle MTS_DISPATCHERS:*
http://docs.oracle.com/cd/A87860_01/doc/server.817/a76961/ch198.htm
- *Oracle Dynamic Performance Views (V\$):*
http://docs.oracle.com/cd/B19306_01/server.102/b14237/dynviews_part.htm#i403961
- *Cygwin X Server:*
<http://www.cygwin.com/>
- *Oracle Net Services: TNS Listener:*
http://docs.oracle.com/html/B13738_01/ch5.htm - i25111
- **Servidores de Aplicaciones Web Tomcat:**
 - *Apache Tomcat:*
<http://tomcat.apache.org/>
 - *Monitoring Apache Tomcat FAQ:*
<http://wiki.apache.org/tomcat/FAQ/Monitoring?highlight=%28CategoryFAQ%29>
 - *Java Management Extensions (JMX) API:*
<http://docs.oracle.com/javase/6/docs/technotes/guides/jmx/>

- *Apache Tomcat Metrics:*
http://docs.oracle.com/cd/E11857_01/em.111/e10115/middlewar_e_apache_tomcat.htm
- *WGET for Windows:*
<http://gnuwin32.sourceforge.net/packages/wget.htm>

5.- TERCERA FASE

El objetivo de este capítulo final es el de analizar la capacidad nativa de *Pandora FMS Open Source*¹⁸ para analizar *SLAs* (*Service Level Agreements*) y generar informes que avalen dichos análisis. Adicionalmente, se analizará la *API* de extensiones de *Pandora FMS*, con la que se podrían integrar soluciones de terceros que cubran las carencias de las utilidades ofrecidas por la plataforma.

5.1.- Marco de Uso

Las situaciones más comunes en las que la generación de informes resulta útil son las siguientes:

- Se desea analizar el grado de cumplimiento de un *SLA* acordado con un cliente sobre un servicio que ofrece la empresa¹⁹.
- Se desea analizar la calidad de un servicio que la empresa esté consumiendo.
- Se desea analizar el estado de máquinas y servicios a lo largo del tiempo, con el fin de deducir las posibles causas de un fallo informático.
- Se desea analizar el estado de máquinas y servicios periódicamente, para poder detectar interrupciones o cortes en los servicios que no hayan sido percibidos *in situ*.

Las soluciones de las siguientes secciones serán analizadas teniendo en cuenta el marco descrito, realizando, si se considera oportuno, apreciaciones sobre la conveniencia o inconveniencia de sus características y funcionalidades concretas.

5.2.- Capacidad Nativa de Reporting

Este capítulo pretende ofrecer un enfoque más profundo y orientado a la práctica que el análisis realizado en la segunda fase de proyecto.

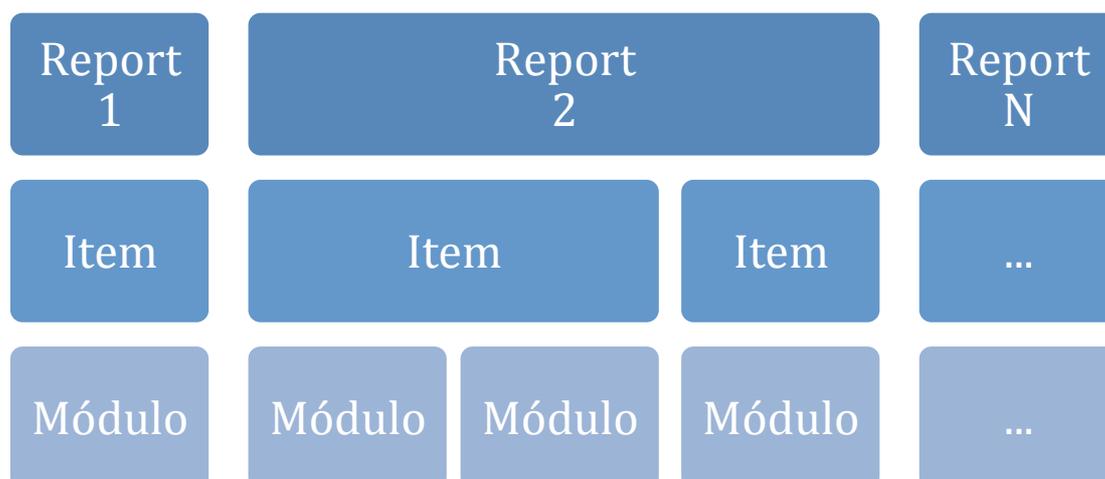
5.2.1.- Estructura de Reporting

Pandora FMS permite crear (y almacenar para su uso futuro) informes o *reports* personalizados que reflejen el estado de agentes o módulos en un periodo de tiempo determinado.

¹⁸ La versión *Enterprise*, como es de esperar, ofrece un mayor número de funcionalidades en cuanto a generación de informes y análisis de *SLAs*. A lo largo de este capítulo, únicamente profundizaremos en el análisis de utilidades de código abierto, o al menos, gratuitas.

¹⁹ Aunque puedan resultar útiles en el ámbito interno de la empresa y de cara al cliente, resulta evidente que estos informes no tendrán validez legal. Los datos extraídos por los módulos se almacenan en la base de datos de *Pandora FMS* y son fácilmente manipulables. En general, la irrefutabilidad de este tipo de informes se antoja muy complicada de alcanzar y su análisis se encuentra totalmente fuera del ámbito de este proyecto.

El sistema de *reporting* de *Pandora* se basa en la siguiente estructura:



- **Report** o informe: Agrupación de más alto nivel que puede exportarse en distintos formatos para dar lugar al producto final. Su contenido se almacena en *MySQL* junto con el resto de datos de la instancia de *Pandora FMS*.

Los *reports* creados se encuentran accesibles bajo la pestaña "*Reporting*" de la consola web de *Pandora*, permitiendo que puedan ser reutilizados para generar distintos documentos en distintos periodos de tiempo sobre los mismos datos.

- Cada *report* se compone de uno o más **Items** de distintos tipos (los indicados en "[3.1.6.2.- Generación de Informes](#)"): gráficas, valores máximos, *MTBF*, ...

El caso de los ítems *SLA* es especial y se analiza con en el próximo apartado.

- Los ítems emplearán los valores de uno o más **módulos** de *Pandora FMS*, cuyo histórico haya sido almacenado.

Esto significa que, en última instancia, el proceso de generación de un informe implica acceder a la base de datos y consultar los resultados que los módulos han ido mostrando a lo largo del tiempo. Por tanto, resulta evidente que aquellos módulos de los que no se esté almacenando información histórica no podrán ser objeto de *reporting*.

5.2.2.- Análisis de Cumplimiento de SLAs

Los *SLAs* son ítems que pueden estar contenidos dentro de un *report* y que, a diferencia del resto de ítems, poseen un estado *booleano*: superado o no superado.

Cada *SLA* posee una o más condiciones, compuestas de:

- Valor mínimo que debe indicar el módulo.

- Valor máximo que debe indicar el módulo.
- Porcentaje de cumplimiento.

Por ejemplo, podemos definir un *SLA* que establezca que la pérdida de paquetes en una interfaz (indicada por un módulo) debe ser inferior al 1% en un 99% de las veces²⁰.

Genéricamente, un acuerdo *SLA* puede componerse de más de una condición: en estos casos, *Pandora FMS* permite especificar si el estado general del *SLA* será el peor o el mejor de los estados de cada condición. En cualquiera de los dos casos, los informes reflejarán qué condiciones han sido cumplidas o incumplidas y con qué porcentaje de cumplimiento.

5.2.3.- Formatos de Salida

5.2.3.1.- Exportación de Reports

Los informes únicamente pueden exportarse en formato *HTML* o *XML* (la versión Enterprise incluye *PDF* y *CSV*).

El formato *HTML* tiene un aspecto cuidado, simple y representativo. En el siguiente ejemplo, exportamos un informe de prueba muy simple que contiene un acuerdo *SLA* con tres condiciones:

S.L.A.		(5 hours) From: May 29, 2013, 9:13 am To: May 29, 2013, 4:00 am				
Agent	Module	Max/Min Values	SLA Limit	SLA Compliance	Status	Criticality
Oracle Kanpoko Txinzer	Procesos Activos	50.00/0.00	99.00%	100%	OK	●
Oracle Kanpoko Txinzer	Transacciones en Curso	1.00/0.00	99.00%	100%	OK	●
Oracle Kanpoko Txinzer	DB Block Buffer Cache Hit Ratio	100.00/86.00	99.00%	0%	Fail	●

No obstante, el informe *HTML* se encuentra contenido en la consola web de *Pandora FMS*, con lo cual el proceso para generar un documento separado se complica notablemente y no resulta trivial.

5.2.3.2.- Exportación de Información de Módulos

La información histórica recolectada por los módulos puede exportarse masivamente en formato *CSV*, *XLS*, y tabla *HTML*, además de poder accederse directamente mediante consultas *SQL* al servidor de base de datos que actúe de *Back-End* de *Pandora FMS*.

Descubrimos las siguientes rutas, dentro de *CentOS*, que apuntan a los ejecutables utilizados por *Pandora FMS* para exportar la información de la base de datos a los distintos formatos:

```
/var/www/html/pandora_console/operation/agentes/exportdata.php
```

²⁰ Nótese el impacto de la frecuencia de chequeo de un módulo sobre la precisión en el grado de cumplimiento de un *SLA*.

```
/var/www/html/pandora_console/operation/agentes/exportdata.excel.php
/var/www/html/pandora_console/operation/agentes/exportdata.csv.php
/var/www/html/pandora_console/operation/agentes/export_csv.php
/var/www/html/pandora_console/operation/events/export_csv.php
```

Es posible que esta información nos resulte de utilidad en secciones posteriores de este documento para integrar soluciones de *reporting* terceros.

5.3.- API Externa de Pandora FMS

La API externa de Pandora FMS es accesible vía HTTP(S) sobre la aplicación PHP hallada en:

```
https://servidor/include/api.php
```

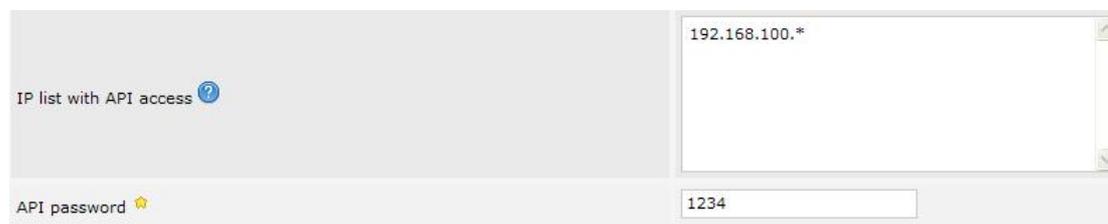
Esta aplicación admite parámetros incrustados en la URL (método GET) y es el método estándar para integrar aplicaciones de terceros en Pandora FMS.

5.3.1.- Consideraciones de Seguridad

El acceso a la API externa está protegido mediante múltiples restricciones de acceso, listadas a continuación:

- Credenciales (User/Password) de Pandora FMS de una cuenta con privilegios suficientes para realizar las operaciones deseadas (en el caso de *reporting*, bastará con privilegios de solo lectura).
- Contraseña de acceso a la API.
- Lista blanca de direcciones IP origen con permiso de acceso a la API.

Estas dos últimas opciones deben estar debidamente definidas para habilitar el acceso a la API. Puede llevarse a cabo desde la misma interfaz web de Pandora:



The screenshot shows a web form with two main input fields. The first field is labeled 'IP list with API access' and contains the text '192.168.100.*'. The second field is labeled 'API password' and contains the text '1234'. There are also small icons (a question mark and a star) next to the labels.

Todos estos parámetros deberán incrustarse en la URL²¹ de la siguiente manera, para el usuario "test" con contraseña "pandora" y siendo "1234" la contraseña de acceso a la API:

```
api.php?apipass=1234&user=test&pass=pandora
```

²¹ La documentación oficial de Pandora FMS insiste en el uso de HTTPS para proteger la confidencialidad de las contraseñas. Aún así, el método GET es considerado inseguro, ya que las trazas de ejecución o el historial del navegador web almacenarán las contraseñas en texto claro, entre otras debilidades.

5.3.2.- Parámetros GET Adicionales

Además de los parámetros "*apipass*", "*user*" y "*pass*", la aplicación admite un gran número de parámetros que permiten realizar consultas y acciones sobre la instancia objetivo. Los más significativos son los siguientes:

- "**op**": Parámetro obligatorio que especifica el tipo de llamada a la API: *get*, *set* o *help*.
- "**op2**": Identificador que, junto con el parámetro "*op*", indica la operación interna a realizar. Como ejemplos, podríamos realizar la operación "*get all_agents*" (*op=get&op2=all_agents*) o "*set events*" (*op=set&op2=events*).
- "**id**", "**id2**", "**other**": Parámetros de la operación, si fueran necesarios. En el caso de existir más de tres parámetros, "*other*" agruparía los argumentos serializados del tercero en adelante.
- "**other_mode**": Formato de la serialización de los parámetros.
- ...

5.3.3.- Llamadas de Prueba

En esta sección se realizarán llamadas "*get*" de prueba a la API externa de *Pandora FMS*, que devolverán un resultado en formato *string* o CSV (*Comma Separated Values*) que podrá ser tratado por un *script* o una aplicación ajena a *Pandora*. El resultado final podría ser, entre otros muchos, un informe de análisis del grado de cumplimiento de un *SLA* determinado.

A continuación se muestran las consultas de prueba realizadas a través del navegador web:

5.3.3.1.- Eventos del Sistema

Consultamos los últimos eventos sucedidos en *Pandora FMS*, llamando a la siguiente URL:

```
http://pandoraserver/pandora_console/include/api.php?op=get&op2=events&return_type=csv&apipass=1234&user=test&pass=pandora
```

Donde los parámetros reseñables son:

- **op=get&op2=events**: Operación de consulta de los últimos 5 (por defecto) eventos de *Pandora FMS*.

El *output* incluye *ID* del evento, *ID* del agente, estado anterior del agente, *ID* del módulo, estado anterior del módulo, *timestamp*, descripción, *URLs* de imágenes utilizadas por *Pandora* para representar el error, ...

A continuación se muestra una de las cinco entradas mostradas:

```
2183;24;0;12;0;2013-06-04 09:12:09;Module App sample (CRITICAL: App "sample" no lanzada ) is going to
```

CRITICAL;1370329929;going_up_critical;425;0;4;;;Tomcat
 Pandoraserver;Applications;applications;Going down to critical
 state;http://192.168.100.173/pandora_console/images/b_red.png;Critical;http://192.168.100.173/pandora_console/images/status_sets/default/severity_critical.png

5.3.3.2.- Todos los Agentes

Para obtener el listado completo de agentes activos en la plataforma, realizamos una llamada a la siguiente *URL*:

```
http://pandoraserver/pandora_console/include/api.php?op=get&op2=all_agents&return_type=csv&apipass=1234&user=test&pass=pandora
```

Donde se indican los parámetros:

- **op=get&op2=all_agents**: Operación para consultar todos los agentes activos en *Pandora FMS*.
- **return_type=csv**: Formato CSV.
- **apipass=1234&user=test&pass=pandora**: Credenciales de autenticación.

El *output* es el siguiente (*ID* del agente; Nombre del agente; Dirección *IP* [ocultada]; Descripción; Grupo):

```
1;pandoraserver;;Created by pandoraserver;Linux
7;BreezeNET_BU1;X.X.X.X;Antena BreezeNET Base Unit 1;Network
8;BreezeNET_RB1;X.X.X.X;Antena BreezeNET Remote Bridge 1;Network
13;BreezeULTRA_BU1;X.X.X.X;Antena BreezeULTRA Base Unit 1;Network
14;BreezeULTRA_RB1;X.X.X.X;Antena BreezeULTRA Remote Bridge 1;Network
15;MACH1000_1;X.X.X.X;Switch Hirschmann MACH1000 Full 16 GE;Network
16;MACH102_1;X.X.X.X;Switch Hirschmann MACH102-8TP-R;Network
17;MACH1000_3;X.X.X.X;Switch Hirschmann MACH1000 Full 16 GE;Network
18;HP_ProLiant_DL380G7_1;X.X.X.X;Servidor HP ProLiant DL380 G7 #1;Embedded
19;HP_ProLiant_DL360G5_1;X.X.X.X;Servidor HP ProLiant DL360 G5 #1;Embedded
20;EMC_VNXe3100_1;X.X.X.X;Cabina EMC VNXe 3100 #1;Embedded
```

21;VMware_ESXi5.1_1;X.X.X.X;VMware ESXi 5.1 sobre HP ProLiant DL360 G5;VMware

22;Oracle Kanpoko Txinzer;X.X.X.X;Oracle 11G R2;Linux

23;Tomcat Kanpoko Txinzer;X.X.X.X;Servidor Tomcat 5;Web Server

24;Tomcat Pandoraserver;localhost;Servidor Tomcat 6;Web Server

5.3.3.3.- Módulos de un Agente

Tomando el *ID* de un agente como referencia, llamamos a la siguiente *URL*:

```
http://pandoraserver/pandora_console/include/api.php?op=get&op2=agent_modules&return_type=csv&other=22&other_mode=url_encode_separator_|&apipass=1234&user=test&pass=pandora
```

Donde se indican los parámetros:

- ***op=get&op2=agent_modules***: Operación de consulta de módulos de un agente
- ***other=22&other_mode=url_encode_separator_|***: El identificador numérico del agente es 22 (*Oracle Kanpoko Txinzer*). El separador de argumentos es el carácter '|', aunque no ha sido necesario utilizarlo.

El resultado es el siguiente (*ID* del agente; *ID* del módulo; Nombre del módulo):

22;404;Errores ORA

22;405;Dictionary Cache Hit Ratio

22;406;Library Cache Hit Ratio

22;407;DB Block Buffer Cache Hit Ratio

22;408;Latch Hit Ratio

22;409;Disk Sort Ratio

22;410;Rollback Segment Wait Ratio

22;411;Dispatcher Workload

22;412;Sesiones Activas

22;413;Procesos Activos

22;414;Transacciones en Curso

22;415;Listener Estado

5.3.3.4.- Información de un Módulo

Tomamos como referencia el identificador de un módulo concreto y llamamos a la *URL*:

```
http://pandoraserver/pandora_console/include/api.php?op=get&op2=module_data&id=411&other=;|604800&other_mode=url_encode_separator_|&apipass=1234&user=test&pass=pandora
```

Significando los parámetros:

- ***op=get&op2=module_data&id=411***: Operación para obtener toda la información recabada por el módulo de *ID 411 (DB Block Buffer Cache Hit Ratio)*.
- ***&other=;|604800&other_mode=url_encode_separator_|***: El carácter ';' indica el carácter separador a utilizar en el CSV de salida. '604800' indica la antigüedad máxima de la información a mostrar en segundos (604800 segundos equivale a una semana).

El *output (Unix Timestamp; Valor)* recortado es el siguiente, mostrado de menor a mayor antigüedad:

```
1370332365;96.82 1370330637;96.83 1370330316;96.82
1370326898;96.83 1370323093;96.84 1370319625;96.85
1370314260;96.86 1370310462;96.87 1370305088;96.88
```

(...)

```
1369798886;85.53 1369797625;85.52 1369797310;85.55
1369796049;85.54 1369795730;85.53 1369794144;85.56
1369792873;85.58 1369792237;85.57 1369791922;85.60
```

5.3.3.5.- Gráficos

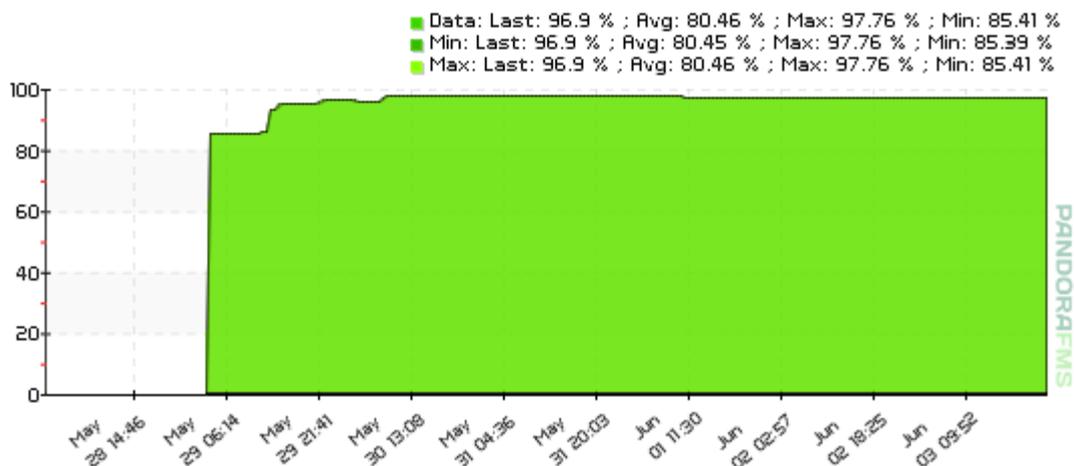
De nuevo, tomando un módulo como referencia, generamos un gráfico que representa los valores que toma el módulo a lo largo de un intervalo de tiempo. La *URL* empleada es la siguiente:

```
http://pandoraserver/pandora_console/include/api.php?op=get&op2=graph_module_data&id=407&other=604800|555|245|PROBANDO|2013-06-04&other_mode=url_encode_separator_|&apipass=1234&user=test&pass=pandora
```

Siendo los parámetros:

- ***op=get&op2=graph_module_data&id=407***: Operación para generar un gráfico en formato *PNG* sobre el módulo de *ID 407* (de nuevo, *DB Block Buffer Cache Hit Ratio*).
- ***other=604800|555|245|PROBANDO|2013-06-04***: Indica el intervalo de tiempo a representar (una semana), los píxeles de anchura-altura (555x245), el nombre del gráfico ("PROBANDO") y la fecha límite (6 de junio del 2013).

El resultado es el siguiente:



Comprobamos que los valores representados se ajustan a los listados en formato (*Unix Timestamp*; Valor): no se tiene información anterior al 29 de mayo, y los primeros datos muestran valores cercanos al 85%. Posteriormente el *DB Block Buffer Cache Hit Ratio* crece y se estabiliza en torno al 96-97%.

Referencias/Enlaces del Capítulo

***NOTA:** Los enlaces han sido comprobados por última vez el día 05/06/2013. Es posible que algunos de ellos estén rotos o hayan sido actualizados. En tal caso, conviene acceder al directorio raíz o página inicial e iniciar la búsqueda de la referencia.

- *Pandora FMS Reports:*
http://www.openideas.info/wiki/index.php?title=Pandora:Documentation_en:Data_Presentation#Reports
- *Pandora FMS Development and Extension:*
http://www.openideas.info/wiki/index.php?title=Pandora:Documentation_en:Development_and_Extension#Reports
- *Pandora FMS External API:*
http://www.openideas.info/wiki/index.php?title=Pandora:Documentation_en:Annex_ExternalAPI#Pandora_FMS_External_API
- *Unix Time Stamp:*
<http://www.unixtimestamp.com/index.php>

6.- CONCLUSIONES

Tras lo visto en el transcurso del proyecto, y con cierta experiencia en otros entornos de monitorización, podemos definir este paradigma de monitorización como la interacción de los siguientes elementos:

- Existe un dispositivo o **sistema objetivo**, que se desea monitorizar.
- Se instalan (o vienen instalados de serie) un conjunto de **sensores o agentes**, ya sean *hardware*, *software*, o una combinación de ellos. Su misión es extraer información en tiempo real y dejarla disponible.
- La **plataforma de monitorización** recabará la información mencionada (de forma activa o pasiva) y la procesará para mostrarla, disparar alertas, o realizar cualquier acción para la que esté preparada.

En este último aspecto, cabe destacar que hoy en día la monitorización no se entiende simplemente como una forma de conocer el estado de un sistema, sino también como una **forma automática de reacción** ante dicho estado: relanzar un proceso si éste no se encuentra activo, liberar memoria de un disco que esté al límite de su capacidad, etc.

Desde el punto de vista general de este paradigma, se observa que **teóricamente es posible monitorizar cualquier parámetro presente** que se nos pueda ocurrir. La auténtica limitación se encuentra en la complejidad práctica de llevar a cabo la monitorización: quizás no exista un sensor *hardware* adecuado en el mercado, o quizás el desarrollo *software* necesario resulta demasiado costoso.

En otras palabras, podemos presumir que **este paradigma seguirá siendo válido por mucho que avance la tecnología**: dispositivos y sensores más sofisticados, líneas de comunicación más complejas, plataformas de monitorización más inteligentes, ...

En el entorno concreto del proyecto, para desarrollar las soluciones requeridas por la empresa, ha resultado muy útil el **apoyo de la "comunidad Open Source"**, donde se ponen en común multitud de problemas habituales y cómo han sido resueltos, además de piezas de *software* de gran utilidad.

Además, se ha podido apreciar que, en un sistema real, no basta con que la monitorización funcione correctamente: debe ser **limpia, y deseablemente, eficiente**. Algunos de estos aspectos deseables son:

- **Poco intrusiva**: evita agentes *software* e instalaciones complejas que lastren el sistema, en medida de lo posible.

- **Minimiza el consumo de recursos**, estableciendo una periodicidad adecuada para cada chequeo y evitando aquellos que saturan la red innecesariamente.
- **No sobrecarga de alertas** al responsable de tratarlas. Debe encontrarse un equilibrio en las alertas: ni debe omitirse información crítica, ni debe alertarse de demasiados eventos. Distinguir entre qué información es crítica y cual no lo es puede ser un dilema más complejo de lo que pueda parecer a primera vista.
- **Escala fácilmente** y da facilidades para monitorizar nuevos dispositivos similares a algunos que ya se estén monitorizando. Diseñar *templates* puede ser un buen punto de partida en este aspecto.
- Ordenada, con una **nomenclatura normalizada** (si es a nivel de empresa u organización, mejor). Esto ayudará a entender qué se está monitorizando en cada dispositivo y facilitará la detección y solución de errores.
- Deja la plataforma de monitorización en un **estado estable**, encargándose de limpiar periódicamente los ficheros temporales o procesos en ejecución derivados de un chequeo en la máquina. El objetivo es que el servidor no llegue a colapsarse por llenado de memoria o saturación de *CPU*. Una primera aproximación sería emplear herramientas como *RRDD-Tool* u otras tareas programadas específicas.

Centrándonos en *Pandora FMS*, donde se monitoriza y se alerta del mal funcionamiento de distintas máquinas que operan con varios niveles de redundancia, surgen las preguntas: ¿Y si se cae el propio *Pandora FMS Server*? ¿Quién me alerta si se cae el propio servidor de correo de las alertas?

Las respuestas a estas preguntas podrían constituir un siguiente paso, no abarcado en este proyecto, que sería **configurar *Pandora FMS* en HA (High Availability)** con varios servidores, *pollers*, y vías de alerta que resultaran en un sistema más robusto, más tolerante a fallos y más escalable.

También se ha observado que *Pandora FMS* **no es una aplicación robusta en términos de seguridad**, ya que no sigue las buenas prácticas de diseño más habituales (cifrado de conexiones, almacenamiento de contraseñas en texto claro, ...) en aplicaciones web. La securización de una instancia de *Pandora FMS* podría dar lugar a un proyecto muy extenso, que involucraría tanto al código de *Pandora FMS* como a la configuración de los dispositivos y sistemas monitorizados.

En definitiva, este proyecto ha resultado de gran utilidad no solo para profundizar en aspectos de monitorización con una plataforma a priori desconocida, sino que también ha ayudado a conocer equipamientos, protocolos y nuevas tecnologías implementadas en entornos empresariales reales.

ANEXO 1.- PLAN DE GESTIÓN

A1.1.- Equipo de Proyecto

El proyecto será abordado de manera individual por el alumno **Cid Rivas, Ander**, quien cumplirá las labores de gestión y ejecución del mismo, siendo el único responsable del éxito del proyecto.

A1.2.- Alcance del Proyecto

A1.2.1.- Producto Objetivo

- Módulos software integrados en la plataforma de monitorización *Pandora FMS Open Source* que permitan la recogida y explotación de información de distintos dispositivos en red.
- Solución que permita, por un lado, analizar el cumplimiento de los *SLA's* contratados por los clientes; y por otro, generar informes que lo avalen.
- Documentación final que describa y contextualice el trabajo realizado.

A1.2.2.- Grupo de Interés

Stakeholder	RoI PFG	Cargo Real	Necesidades/Expectativas
Cid Rivas, Ander	Realizador PFG	Alumno 4º curso GII, Especialidad Computadores y Sistemas, UPV/EHU	<ul style="list-style-type: none"> - Superación de PFG: visto bueno del Tribunal. - Visto bueno del trabajo por parte de Empresa y Director. - Ser dotado de equipamiento y la información específica necesaria para el desarrollo del PFG por parte de Empresa
Aguirre Zulaika, Mikel	Representante de Empresa	Director de Departamento Soluciones, COM&MEDIA	<ul style="list-style-type: none"> - Nula o mínima intromisión en la actividad natural de la empresa. - Obtención de soluciones con la mayor calidad posible. - Independencia y capacidad de autoformación del alumno. - Percepción de esfuerzo y eficiencia del alumno (<i>Feedback</i> periódico). - Responsabilidad del

			alumno frente a trámites administrativos.
Miguel Alonso, José	Director PFG	Docente FISS, Coordinador de Especialidad Computadores y Sistemas	<ul style="list-style-type: none"> - Correcta interpretación y desarrollo del proyecto por parte del alumno. - <i>Feedback</i> periódico por parte del alumno. - Situar el PFG en el marco de la Administración de Redes y Sistemas. - Responsabilidad del alumno frente a trámites administrativos.
	Tribunal PFG		<ul style="list-style-type: none"> - Satisfacción de Empresa y Director. - Demostración de que el evaluado ha asimilado las competencias aplicables propias de la titulación²² - El esfuerzo realizado por el alumno cumple con el mínimo estipulado para un PFG (≈ 300 horas). - El desarrollo del Proyecto se ajusta al Plan de Gestión entregado por el mismo alumno.

A1.2.3.- Restricciones Generales

- No se contemplan aquellas soluciones potenciales a los problemas planteados que impliquen un coste económico directo en concepto de adquisición de software o licencias. Se excluyen aquellos gastos contemplados o aprobados por el Tutor Externo.
- Los cambios, pruebas y demás acciones fruto del desarrollo del proyecto no deben afectar a la continuidad de la actividad de la empresa.
- La documentación se hallará bajo *Licencia Creative Commons BY-NC-ND*, salvo en aquellos puntos en los que se especifique lo contrario y aquellas piezas de software que entren en el ámbito del Convenio.

A1.2.4.- Costes Financieros

No se contempla ningún coste financiero directamente imputable al desarrollo del Proyecto, tal y como se expresa en las Restricciones Generales.

²² http://www.ehu.es/p200-content/es/pls/entrada/plew0040.htm_siguiente?...

A1.3.- Plan de Comunicaciones

A1.3.1.- Comunicaciones Externas

Miguel Alonso, José (Director de Proyecto)

- *Feedback* periódico. Dejar disponibles las versiones parciales del proyecto en un sitio web gratuito dedicado (En *Google Sites*: <https://sites.google.com/site/pfgandercid/>)
- Datos de contacto:
 - Email: j.miguel@ehu.es
 - Teléfono Despacho: 943 01 80 19
 - N° Despacho:

Aguirre Zulaika, Mikel (Tutor Externo de Empresa)

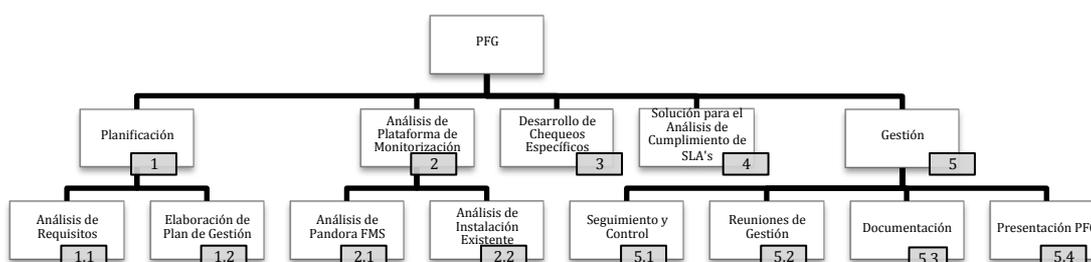
- Reuniones en puesto de trabajo.
- Datos de contacto:
 - Email: maguirre@comymedia.com
 - Teléfono Oficina: 943 31 60 68
 - Móvil: 619 31 60 40

A1.3.2.- Difusión del PFG

El producto final se ajustará a la propiedad y licencia(s) especificada(s) en apartados anteriores.

La memoria del *PFG* constará en el repositorio *ADDI* para Proyectos Fin de Carrera / Proyectos Fin de Grado y será supervisada por el propio Alumno, el Director de Proyecto, el Representante de Empresa y aquellos pertenecientes al Tribunal.

A1.4.- Estructura de Desglose de Trabajo (EDT)



ID	Descripción	Resultado (Directo)
1	Planificación	Plan de Gestión (Anexo)
1.1	Análisis de Requisitos	Anotaciones, Documentos (No Anexos)
1.2	Elaboración de Plan de Gestión	Alcance del Proyecto, Plan de Comunicaciones, EDT, Planificación Temporal, Plan de Riesgos
2	Análisis de Plataforma de Monitorización	Documentación de la Primera Fase

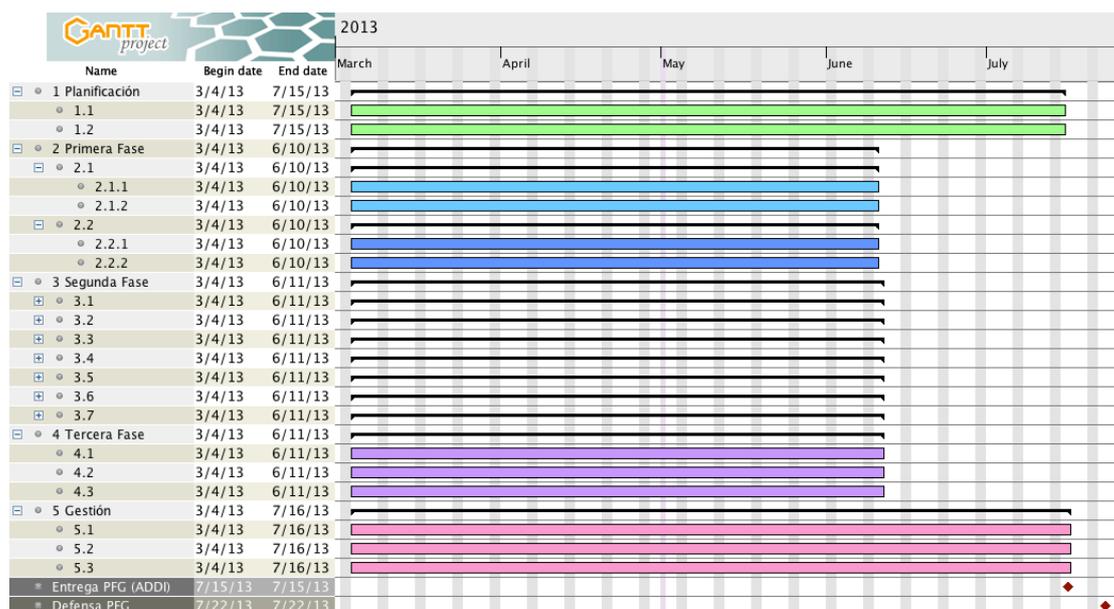
2.1	Análisis de Pandora FMS	Informe de Análisis de Pandora FMS
2.1.1	Análisis de Funcionamiento	
2.1.2	Análisis de Tecnologías Subyacentes	
2.1.3	Análisis de Funcionalidades	
2.2	Análisis de Instalación Existente	Informe de Análisis de la Instalación
2.2.1	Análisis Técnico	Ficha Técnica de la Instalación
2.2.2	Análisis Funcional	Listado de Funcionalidades de la Instalación
3	Desarrollo e Integración de Módulos de Chequeos Específicos	Documentación de la Segunda Fase
3.1	<i>Disp1: Equipamiento WiMAX Alvarion</i>	Documentación del Desarrollo e Integración del Módulo
3.1.1	Análisis Funcional de Dispositivo	Informe de Análisis del <i>Disp1</i>
3.1.2	Desarrollo de Módulos de Recogida de Información	Documentación del Desarrollo del Módulos, Código (Anexo)
3.1.3	Configuración de Alertas/Disparadores	Documentación de Configuración de Alertas/Disparadores
3.1.4	Puesta en Marcha y Evaluación sobre un Dispositivo Operativo	Documentación de la Evaluación
3.2	<i>Disp2: Equipamiento Networking Industrial Hirschmann</i>	Documentación del Desarrollo e Integración del Módulo
3.2.1	Análisis Funcional de Dispositivo	Informe de Análisis del <i>Disp2</i>
3.2.2	Desarrollo de Módulos de Recogida de Información	Documentación del Desarrollo del Módulos, Código (Anexo)
3.2.3	Configuración de Alertas/Disparadores	Documentación de Configuración de Alertas/Disparadores
3.2.4	Puesta en Marcha y Evaluación sobre un Dispositivo Operativo	Documentación de la Evaluación
3.3	<i>Disp3: Servidores Proliant HP</i>	Documentación del Desarrollo e Integración del Módulo
3.3.1	Análisis Funcional de Dispositivo	Informe de Análisis del <i>Disp3</i>
3.3.2	Desarrollo de Módulos de Recogida de Información	Documentación del Desarrollo del Módulos, Código (Anexo)

3.3.3	Configuración de Alertas/Disparadores	Documentación de Configuración de Alertas/Disparadores
3.3.4	Puesta en Marcha y Evaluación sobre un Dispositivo Operativo	Documentación de la Evaluación
3.4	<i>Disp4</i> : Sistemas de almacenamiento <i>EMC²</i>	Documentación del Desarrollo e Integración del Módulo
3.4.1	Análisis Funcional de Dispositivo	Informe de Análisis del <i>Disp4</i>
3.4.2	Desarrollo de Módulos de Recogida de Información	Documentación del Desarrollo del Módulos, Código (Anexo)
3.4.3	Configuración de Alertas/Disparadores	Documentación de Configuración de Alertas/Disparadores
3.4.4	Puesta en Marcha y Evaluación sobre un Dispositivo Operativo	Documentación de la Evaluación
3.5	<i>Disp5</i> : Soluciones de Virtualización <i>VMware</i>	Documentación del Desarrollo e Integración del Módulo
3.5.1	Análisis Funcional de Dispositivo	Informe de Análisis del <i>Disp5</i>
3.5.2	Desarrollo de Módulos de Recogida de Información	Documentación del Desarrollo del Módulos, Código (Anexo)
3.5.3	Configuración de Alertas/Disparadores	Documentación de Configuración de Alertas/Disparadores
3.5.4	Puesta en Marcha y Evaluación sobre un Dispositivo Operativo	Documentación de la Evaluación
3.6	<i>Disp6</i> : SGBD <i>Oracle</i>	Documentación del Desarrollo e Integración del Módulo
3.6.1	Análisis Funcional de Dispositivo	Informe de Análisis del <i>Disp6</i>
3.6.2	Desarrollo de Módulos de Recogida de Información	Documentación del Desarrollo del Módulos, Código (Anexo)
3.6.3	Configuración de Alertas/Disparadores	Documentación de Configuración de Alertas/Disparadores
3.6.4	Puesta en Marcha y Evaluación sobre un Dispositivo Operativo	Documentación de la Evaluación
3.7	<i>Disp7</i> : Servidores <i>Tomcat</i>	Documentación del Desarrollo e Integración del Módulo
3.7.1	Análisis Funcional de Dispositivo	Informe de Análisis del <i>Disp7</i>

3.7.2	Desarrollo de Módulos de Recogida de Información	Documentación del Desarrollo de Módulos, Código (Anexo)
3.7.3	Configuración de Alertas/Disparadores	Documentación de Configuración de Alertas/Disparadores
3.7.4	Puesta en Marcha y Evaluación sobre un Dispositivo Operativo	Documentación de la Evaluación
4	Solución para el Análisis de Cumplimiento de SLA's	Documentación de la Tercera Fase
4.1	Análisis de Capacidad Nativa de Pandora FMS para el Análisis del Grado de Cumplimiento (%) de SLA's	Informe del Análisis
4.2	Análisis de API Externa para Integración de Herramientas de Terceros en Pandora FMS	Informe del Análisis. Posible Documentación de Integración
4.3	Ejecución de Pruebas contra la API Externa	Documentación de las Pruebas
5	Gestión	
5.1	Seguimiento y Control	Documento de Seguimiento y Control (Anexo)
5.2	Reuniones de Gestión	Anotaciones (No Anexas)
5.3	Integración y Formateo de Documentación	Documentación PFG Final
5.4	Presentación de Defensa PFG	Presentación (No Anexa)

A1.5.- Planificación Temporal

A1.5.1- Diagrama de Gantt e Hitos



A1.5.2.- Distribución de Carga por Semanas

ID	MARZO				ABRIL				MAYO				JUNIO			JULIO		TOTAL		
	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17		S18	S19
1.1	4	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5							10
1.2	10	2	1	1	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5				20
2.1.1		5																		5
2.1.2		5																		5
2.1.3		5																		5
2.2.1		5																		5
2.2.2		5	5																	10
3.1.1			5																	5
3.1.2			15																	15
3.1.3				10																10
3.1.4				5																5
3.2.1				10																10
3.2.2					20															20
3.2.3					5	5														10
3.2.4						5														5
3.3.1						5														5
3.3.2						10	10													20
3.3.3							10													10
3.3.4							5													5
3.4.1								5												5
3.4.2								15												15
3.4.3								5	5											10
3.4.4									5											5
3.5.1									5											5
3.5.2									10	5										15
3.5.3										10										10
3.5.4										5										5
3.6.1										5	5									10
3.6.2											15									15
3.6.3											5	5								10
3.6.4												5								5
3.7.1												5								5
3.7.2												10	5							15
3.7.3													10							10
3.7.4													5							5
4.1													5	10						15
4.2														10						10
4.3														5						5
5.1	2	2	1	1	1	1	1	1	1	1	1	1	1	0.5	0.5	0.5	0.5	1	2	20
5.2	1				1					1				1				1		5
5.3															2	2	2	4		10
5.4	2.5	2.5													5	2	1	1	1	15
TOTAL	20	32	28	28	28	27	27	27	28	27	27	27	27	27	6	5	3.5	5	7	405

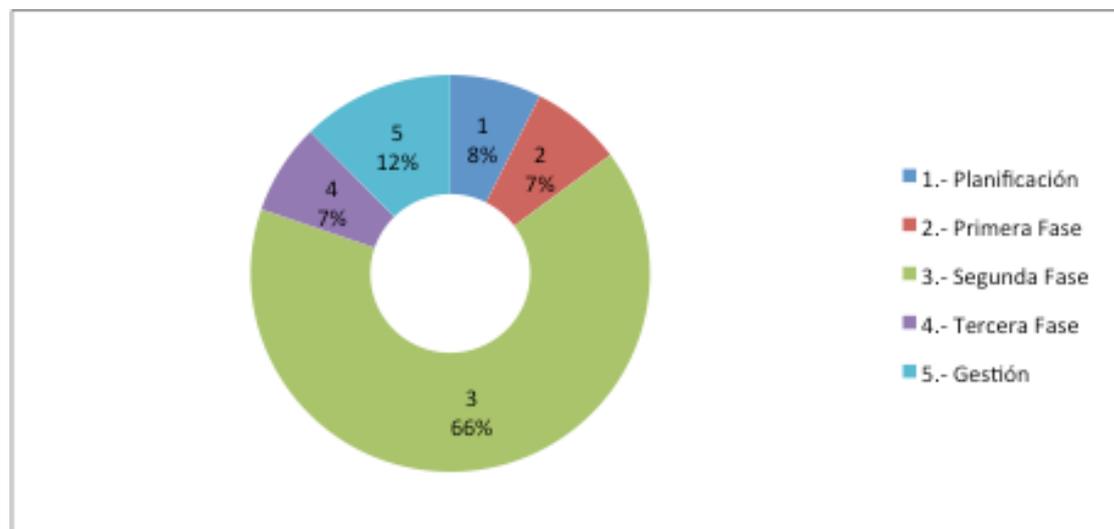


Figura 1: Gráfica de Reparto de Carga por Grupo de Tarea

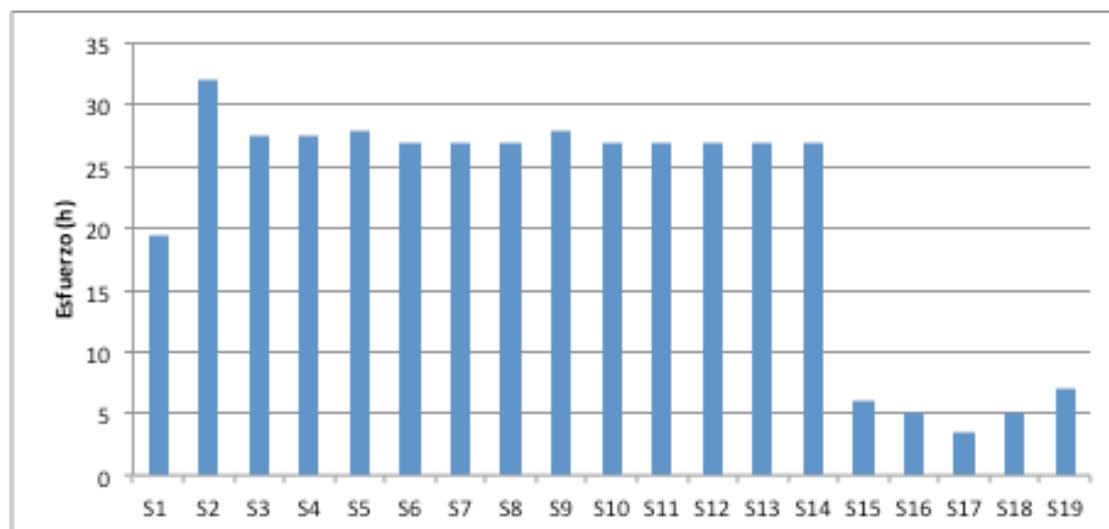


Figura 2: Gráfica de Reparto de Carga por Semana

A1.6.- Plan de Riesgos

ID	Riesgo	Prob.	Impacto	Prevención	Paliación
R1	Baja prolongada (> 6 semanas o críticas) del Alumno	Muy baja	Muy alto		Replanificación integral del PFG. Convocatoria de Septiembre.
R2	Baja (< 6 semanas no críticas) del Alumno	Muy baja	Alto		Replanificación. Desplazamiento del esfuerzo a las semanas S15, S16, S17 y S18. Posible incremento del esfuerzo semanal hasta las 40h. Posible reducción del alcance.
R3	Baja prolongada (> 6 semanas) del Tutor Externo	Muy baja	Medio	Favorecer la independencia del Alumno en la definición y ejecución de tareas.	Búsqueda y relevo de Tutor (suplente) en la empresa. Replanificación de reuniones.
R4	Baja (< 6 semanas no críticas) del Tutor Externo	Muy baja	Bajo	Favorecer la independencia del Alumno en la definición y ejecución de tareas.	Búsqueda de Tutor (suplente) en la empresa. Replanificación de reuniones.
R5	Baja prolongada (> 6 semanas o críticas) del Director	Muy baja	Bajo	Conocimiento previo del Alumno de las labores administrativas a cumplir por el Director	Toma de las responsabilidades administrativas del Director por parte del Alumno. Supresión de reuniones.
R6	Baja (< 6	Muy baja	Bajo	Conocimiento previo	Toma de las

	semanas no críticas) del Director			del Alumno de las labores administrativas a cumplir por el Director	responsabilidades administrativas del Director por parte del Alumno. Replanificación de reuniones
R7	Desvío significativo (>5 horas) en Primera Fase	Baja	Bajo / Medio		Desplazamiento de la carga de trabajo (adelanto o retraso). Posible reducción del alcance del análisis.
R8	Desvío significativo (>5 horas) en Segunda Fase	Alta	Bajo/Medio		Desplazamiento de la carga de trabajo (adelanto o retraso). Posible eliminación/adición de dispositivos.
R9	Desvío significativo (>5 horas) en Tercera Fase	Media	Bajo / Medio	Replanificación de la Tercera Fase una vez realizada la Primera Fase	Desplazamiento de la carga de trabajo.
R10	Coste financiero ineludible en el desarrollo de soluciones	Baja	Bajo / Medio		Consulta al Tutor Externo. Si no se autoriza, descarte de la solución y consiguiente reducción del alcance del proyecto.
R11	Impacto en la continuidad de negocio ineludible en el desarrollo de soluciones	Media	Medio	Desarrollo en entornos virtuales de prueba (mientras sea posible)	Consulta al Tutor Externo. Si no se autoriza, descarte de la solución y consiguiente reducción del alcance del proyecto.
R12	Pérdida de información	Baja	*	Plan de <i>Backup</i> en puesto de trabajo y en equipo propio	Replanificación. Repetición de tareas.
R13	Cambios en las especificaciones (por parte de Director o Tutor Externo)	Baja	*	Definición y acuerdo del alcance del proyecto	Replanificación. Posible desplazamiento de carga de trabajo a las semanas S15, S16, S17 y S18.

A1.6.1.- Plan de Copias de Seguridad

ID	Elementos Respaldados	Periodicidad	Soporte
C1	Puesto de Trabajo COM&MEDIA	Semanal	HDD Externo
C2	Directorio PFG	Diaria	Google Drive
C3	Equipo Personal	Semanal	HDD Externo

ANEXO 2.- INFORME DE SEGUIMIENTO Y CONTROL

A2.1.- Incidencias

ID	Sem.	Tipo	Descripción	Contramedida
I1	S2	Org.	Se desarrollará el PFG sobre un entorno virtual en lugar de una plataforma existente	Replanificación: Grupo de tareas 2.2 (Análisis de Instalación Existente) sustituida por nueva tarea 2.2 (Instalación en Entorno Virtual, 10h)
I2	S3	Téc.	Lentitud y caídas de conexión VPN. No soporta conexiones simultáneas con las mismas credenciales (desde el PC y pandoraserver)	Instalación de <i>NetExtender</i> (cliente VPN) en <i>pandoraserver</i> . Enrutamiento del tráfico de @IP que se encuentren dentro de la VPN a través de <i>pandoraserver</i>
I3	S5	Org.	No se encuentran alertas interesantes para <i>Hirschmann</i>	R8. Decremento de dedicación en Configuración de Alertas en favor de Desarrollo de Módulos
I4	S6	Téc.	Las consultas IPMI no funcionan	Desactivar <i>SELinux</i> sobre <i>pandoraserver</i> . Permitir el tráfico dirigido a <i>UDP/623</i> en <i>IPTables</i>
I5	S7	Téc.	<i>Pandora FMS</i> muestra microcortes en las fuentes de alimentación de los servidores HP que <i>iLO</i> no muestra	Se desconoce la causa.
I6	S7	Téc.	Los identificadores numéricos de los sensores no son consistentes en las distintas generaciones de servidores HP	Se ha creado un <i>plug-in</i> genérico, que recibe el ID numérico del sensor como parámetro
I7	S7	Org.	Las alertas interesantes para HP han consumido 3/10h de la dedicación planificada	R8. Decremento de dedicación en Configuración de Alertas en favor de Desarrollo de Módulos
I8	S8	Org.	Baja del Tutor Externo (28 abril - 6 mayo)	R4. Ninguna desviación significativa en el proyecto
I9	S8	Org.	No se encuentran alertas interesantes para cabinas EMC.	R8. Decremento de dedicación en Configuración de Alertas en favor de Desarrollo de Módulos
I10	S8	Téc.	Borrado accidental de la base de datos MySQL de <i>pandoraserver</i> (25 de abril).	R12. Recuperación total de <i>pandoraserver</i> desde la Copia de Seguridad C1 (22 abril). Reparación de tablas MySQL (<i>repair table tserver@pandora</i>)
I11	S9	Téc.	Los <i>plug-ins</i> EMC no funcionan	R8. Dedicación de 10h extra en Desarrollo de Módulos. Solución técnica: actualización de librerías. Actualización de la variable de entorno <i>\$LD_LIBRARY_PATH</i> . Aceptar los certificados digitales obsoletos de cabinas EMC
I12	S9	Org.	Día del Trabajador (Impacto: 5h)	
I13	S10	Org.	No se encuentran alertas interesantes para entornos VMware.	R8. Adelantamiento de tareas
I14	S10	Téc.	<i>Bug resxtp</i> (Reconocido por VMware Inc: http://blogs.vmware.com/vsphere/)	Método alternativo de extracción de información: conexión SSH y ejecución de <i>esxtp</i> sobre ESXi

			2012/11/resxtop-fails-to-connect-to-a-vsphere-5-1-host.html). No es posible establecer conexiones remotas con servidores que posean certificados no válidos	
I15	S11	Org.	No se encuentran alertas interesantes para <i>SGBD Oracle</i>	R8. Decremento de dedicación en Configuración de Alertas en favor de Desarrollo de Módulos
I16	S11	Téc.	No existen las tablas <i>SYS.MONITOR</i> ni <i>SYS.SYSTEMSTATS</i> indicadas en la documentación de <i>Oracle</i>	El análisis de <i>plug-ins</i> de <i>Nagios</i> revela tablas alternativas con contenidos idénticos. No se encuentra un motivo razonable
I17	S12	Org.	Ausencia por motivos personales, 22 mayo (Impacto: 5h)	
I18	S12	Org.	Exceso de dedicación planificada para servidores <i>Tomcat</i> . Los módulos son relativamente simples y no se encuentran alertas interesantes	R8. Adelantamiento de tareas
I19	S12	Téc.	"Crash" de tablas de <i>Pandora</i> en <i>MySQL</i>	Reparación de tablas <i>MySQL</i> (<i>repair table tserver@pandora</i>)
I20	S13	Téc.	Caídas repetidas de servicios en red y cambios no deseados en <i>/etc/resolv.conf</i>	Restauración manual de ficheros y eliminación de paquetes (<i>yum erase</i>) <i>wpa_supplicant</i> y <i>NetworkManager</i>
I21	S13	Org	No existe ninguna herramienta de <i>reporting</i> ni extensiones directamente integrables en <i>Pandora FMS</i>	R9. Replanificación y modificación del alcance: analizar la <i>API</i> Externa de <i>Pandora FMS</i> y realizar pruebas (tareas 4.2 y 4.3)
I22	S13	Org.	Adelantamiento voluntario de tareas del bloque 5, fuera del horario de oficina	
I23	S14	Org.	Ausencia por motivos personales, 3 de junio (Impacto: 5h)	

A2.2.- Alcance Final

A.2.2.1.- Producto Final

El producto final comprende los siguientes elementos:

- *Scripts* que dan lugar a los *plug-ins* utilizados por los módulos desarrollados. Dichos *scripts* se encuentran anexados a este documento.
- Disco duro virtual (*Pandora FMS Server CentOS.vmdk*) que contiene la instalación completa y operativa del servidor *CentOS* virtual utilizado a lo largo del proyecto.
- Volcado de la base de datos *MySQL* de *pandoraserver* (*pandora.sql*). Es posible importar este volcado en otra instancia de *MySQL* y cargar así los módulos, agentes y demás configuraciones de *Pandora FMS* realizadas en el proyecto.
- Documentación final, bajo el nombre *PFG_Ander_Cid.pdf*.

- Llaverero de direcciones y claves ocultadas en *PFG_Ander_Cid.pdf*, con el fin de no comprometer los sistemas de la empresa.

A.2.2.2.- Costes Financieros

Según lo planificado, no se ha realizado ningún gasto económico directamente imputable al proyecto.

A2.3.- Dedicación y Costes

El seguimiento y control de la dedicación (medida en horas de trabajo) se recoge en el documento *PFG_Plan_Temp.xlsx*, que ha sido actualizado diariamente para dar lugar a la siguiente tabla final:

ID	MARZO												ABRIL						MAYO						JUNIO						JULIO			TOTAL		
	P	R	P	R	P	R	P	R	P	R	P	R	P	R	P	R	P	R	P	R	P	R	P	R	P	R	P	R	P	R	P	R				
1.1	4	4	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	10	10	20	
1.2	10	10	2	2	1	1	1	1	1	1	1	1	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	1.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	20	20	6.5	
2.1.1			5	5.5																													5	5	5	
2.1.2			5	5																													5	5	5	
2.1.3			5	4																													10	12	10	
2.2			5	5																													5	5	5	
3.1.1					5	5																											5	5	15	
3.1.2					15	14																											10	14	10	
3.1.3							10	7																									5	5	7	
3.1.4							5	5																									5	5	10	
3.2.1								10	10																								10	10	10	
3.2.2									20	25																							10	30	20	
3.2.3									5	5																							10	10	0	
3.2.4									5	5																							5	5	5	
3.3.1										5	5																						5	5	5	
3.3.2										10	10																						20	34	20	
3.3.3										10	10																						10	10	3	
3.3.4											5																						5	5	5	
3.4.1											5	5																					5	5	5	
3.4.2											15	13																					15	25	15	
3.4.3											5	5																					5	5	10	
3.4.4											5	5																					5	5	5	
3.5.1											5	3																					5	3	3	
3.5.2											10	9																					15	14	10	
3.5.3											5	5																					10	10	0	
3.5.4											3	3																					5	6	6	
3.6.1											5	5																					10	10	10	
3.6.2											15	20																					15	26	15	
3.6.3											5	5																					5	10	0	
3.6.4											5	5																					5	5	5	
3.7.1											5	5																					5	5	5	
3.7.2											5	4																					15	4	10	
3.7.3											10	4																					10	0	5	
3.7.4											5	5																					5	5	5	
4.1											5	15																					15	15	15	
4.2											5	10																					10	10	10	
4.3											5	5																					5	5	5	
5.1	2	2	2	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	20	20	20	
5.2	1	1																															5	5	5	
5.3																																	10	10	10	
5.4	2.5	2.5	2.5	2.5																													15	15	15	
TOTAL	20	20	27	27	28	28	28	28	28	28	27	27	27	27	27	27	27	27	27	27	32	27	31	6	2.5	5	0.5	3.5	0	5	0	7	0	400	374.5	

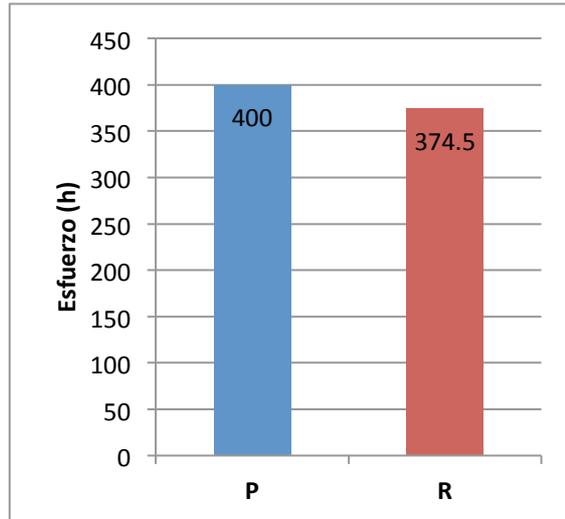


Figura 3: Contraste del Tiempo Planificado y Real Invertido en el Proyecto

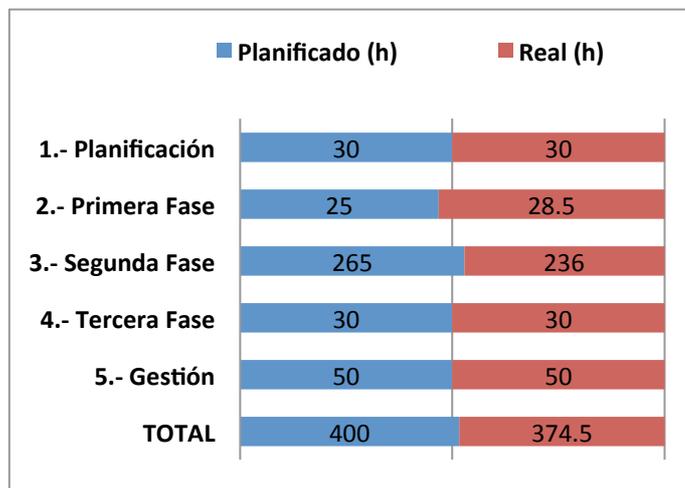


Figura 4: Contraste del Tiempo Planificado y Real Invertido en cada Grupo de Tareas

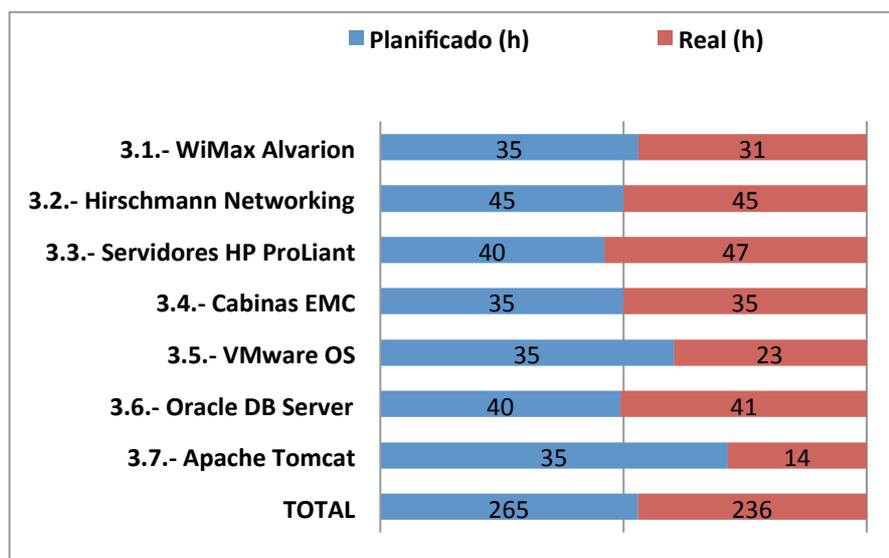


Figura 5: Contraste del Tiempo Planificado y Real Invertido en la Segunda Fase

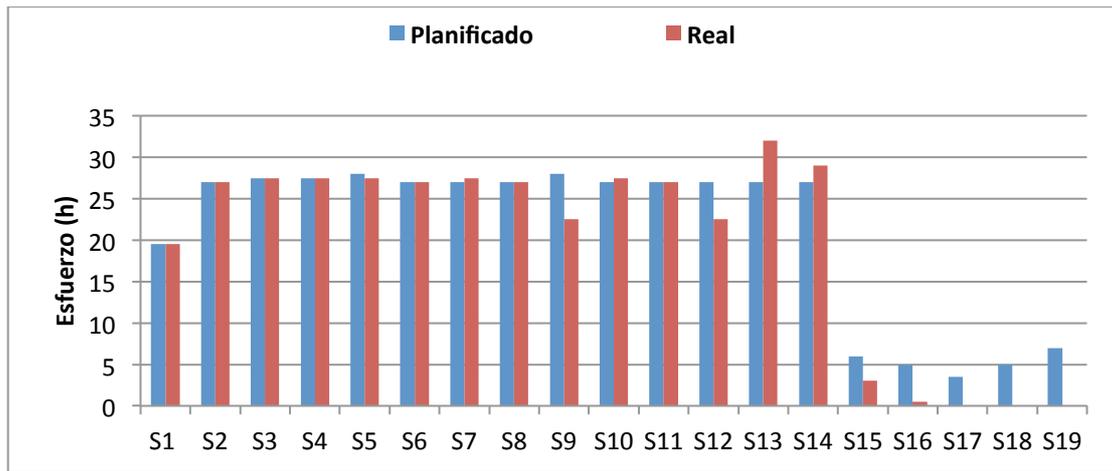


Figura 6: Reparto de Carga Semanal

A2.4.- Adquisiciones y Licencias

ID	Pieza Software	Localización	Licencia
A1	<i>perdidos.pl</i>	http://pandorafms.com/index.php?sec=Library&sec2=repository&lng=es&action=view_PUI&id_PUI=227	Unlicensed
A2	Consultas <i>PyWBEM</i>	http://sourceforge.net/apps/mediawiki/pywbem/index.php?title=Client_Examples	Unlicensed
A3	<i>check_esxcli_harray</i>	http://exchange.nagios.org/directory/Plugins/Hardware/Server-Hardware/HP-(Compaq)/check_esxcli_harray/details	GPLv2
A4	<i>check_oracle_instantclient</i>	http://exchange.nagios.org/directory/Plugins/Databases/Oracle/Check-Oracle-status-%26-health-without-install-Oracle-client/details	Unlicensed

ANEXO 3.- *Scripts Generados*

***NOTA:** Únicamente en los *scripts* propios, se han omitido todas las cabeceras (que indican el nombre del autor, licencia de uso, breve descripción de funcionamiento, dependencias reseñables, ...) a fin de hacer más compacta esta sección.

A3.1.- *Scripts Networking Industrial Hirschmann*

perdidos.pl (original)

```
#!/usr/bin/perl
use Net::SNMP;
use Getopt::Long;
sub procesar_argumentos(){
    $status = GetOptions(
        "h=s" => \$ip,          "ip=s" => \$ip,
        "p=i" => \$interfaz,    "interfaz=i" => \$interfaz,
        "c=s" => \$comunidad,   "comunidad=s" => \$comunidad,
    );
}
$paquetes_perdidos;
$paquetes_entrantes;
@datos_T1;
procesar_argumentos();
$fichero = $ip.'.'.$interfaz.':paq';
if (-e $fichero)
{
    # Ejecucion normal, ya tenemos referencias anteriores
    # Paquetes descartados
    my $oid = '.1.3.6.1.2.1.2.2.1.13.'.$interfaz;
    my $host = $ip;
    my $community = $comunidad;
    my ($session, $error) = Net::SNMP->session(-hostname =>
$host, -community => $community);
    my $result = $session->get_request(-varbindlist =>
[$oid],);
    if (!defined $result){
        printf "ERROR: %s.\n", $session->error();
        $session->close();
        exit 1;
    }
    my $paquetes_descartados = $result->{$oid};
    # Paquetes erroneos
    my $oid = '1.3.6.1.2.1.2.2.1.14.'.$interfaz;
    my $result = $session->get_request(-varbindlist =>
[$oid],);
    if (!defined $result){
        printf "ERROR: %s.\n", $session->error();
        $session->close();
        exit 1;
    }
    my $paquetes_erroneos = $result->{$oid};
    # Paquetes desconocidos
    my $oid = '.1.3.6.1.2.1.2.2.1.15.'.$interfaz;
```

```

    my $result = $session->get_request(-varbindlist =>
[$oid],);
    if (!defined $result){
        printf "ERROR: %s.\n", $session->error();
        $session->close();
        exit 1;
    }
    my $paquetes_desconocidos = $result->{$oid};
    $paquetes_perdidos = $paquetes_descartados +
$paquetes_erroneos + $paquetes_desconocidos;

    # Paquetes entrantes unicast
    my $oid = '.1.3.6.1.2.1.2.2.1.11.'.$interfaz;
    my $result = $session->get_request(-varbindlist =>
[$oid],);
    if (!defined $result){
        printf "ERROR: %s.\n", $session->error();
        $session->close();
        exit 1;
    }
    my $paquetes_unicast = $result->{$oid};

    # Paquetes entrantes no-unicast
    my $oid = '.1.3.6.1.2.1.2.2.1.12.'.$interfaz;
    my $result = $session->get_request(-varbindlist =>
[$oid],);
    if (!defined $result){
        printf "ERROR: %s.\n", $session->error();
        $session->close();
        exit 1;
    }
    my $paquetes_no_unicast = $result->{$oid};
    $session->close();
    $paquetes_entrantes = $paquetes_unicast +
$paquetes_no_unicast;
    open (fichero, $fichero) || die "no se puede abrir el
fichero.\n";
    my $line = <fichero>;
    close fichero;
    print @line;
    @datos_T1 = split(/;/, $line);
}
else
{
    # Es la primera ejecucion
    # Paquetes descartados
    my $oid = '.1.3.6.1.2.1.2.2.1.13.'.$interfaz;
    my $host = $ip;
    my $community = $comunidad;
    my ($session, $error) = Net::SNMP->session(-hostname =>
$host, -community => $community);
    my $result = $session->get_request(-varbindlist =>
[$oid],);
    if (!defined $result){
        printf "ERROR: %s.\n", $session->error();
        $session->close();
        exit 1;
    }

```

```

    }
    my $paquetes_descartados = $result->{$oid};
    # Paquetes erroneos
    my $oid = '.1.3.6.1.2.1.2.2.1.14.'. $interfaz;
    my $result = $session->get_request(-varbindlist =>
[$oid],);
    if (!defined $result){
        printf "ERROR: %s.\n", $session->error();
        $session->close();
        exit 1;
    }
    my $paquetes_erroneos = $result->{$oid};
    # Paquetes desconocidos
    my $oid = '.1.3.6.1.2.1.2.2.1.15.'. $interfaz;
    my $result = $session->get_request(-varbindlist =>
[$oid],);
    if (!defined $result){
        printf "ERROR: %s.\n", $session->error();
        $session->close();
        exit 1;
    }
    my $paquetes_desconocidos = $result->{$oid};
    $paquetes_perdidos = $paquetes_descartados +
$paquetes_erroneos + $paquetes_desconocidos;
    # Paquetes entrantes unicast
    my $oid = '.1.3.6.1.2.1.2.2.1.11.'. $interfaz;
    my $result = $session->get_request(-varbindlist =>
[$oid],);
    if (!defined $result){
        printf "ERROR: %s.\n", $session->error();
        $session->close();
        exit 1;
    }
    my $paquetes_unicast = $result->{$oid};
    # Paquetes entrantes no-unicast
    my $oid = '.1.3.6.1.2.1.2.2.1.12.'. $interfaz;
    my $result = $session->get_request(-varbindlist =>
[$oid],);
    if (!defined $result){
        printf "ERROR: %s.\n", $session->error();
        $session->close();
        exit 1;
    }
    my $paquetes_no_unicast = $result->{$oid};
    $paquetes_entrantes = $paquetes_unicast +
$paquetes_no_unicast;
    $session->close();
    open(fichero, '>'. $fichero) || die "no se puede abrir el
fichero\n";
    push (@datos, $paquetes_perdidos);
    push (@datos, $paquetes_entrantes);
    my @registro = join (";", @datos);
    print fichero "@registro\n";
    close fichero;
    print 0;
    exit 0;
}

```

```

    open (fichero, '>'.$fichero) || die "no se puede abrir el
fichero.\n";
    push (@datos, $paquetes_perdidos);
    push (@datos, $paquetes_entrantes);
    my @registro = join (";", @datos);
    print fichero "@registro\n";
    close fichero;
    # Calculamos el porcentaje de perdidas
    $incremento_perdidos = $paquetes_perdidos - $datos_T1[0];
    $incremento_entrantes =
$paquetes_entrantes - $datos_T1[1];
    #Evitar divisiones por cero
    if ($incremento_entrantes != 0) {
        $perdidos = $incremento_perdidos /
$incremento_entrantes;
    } else {
        $perdidos = 0;
    }
    if ($perdidos < 0) { print 0; }
    else { print $perdidos; }

```

perdidos_s.pl

***NOTA:** Los cambios introducidos (respecto al script original) se han remarcado en **negrita y subrayado**.

```

#!/usr/bin/perl
use Net::SNMP;
use Getopt::Long;
sub procesar_argumentos(){
    $status = GetOptions(
        "h=s" => \$ip,          "ip=s" => \$ip,
        "p=i" => \$interfaz,   "interfaz=i" => \$interfaz,
        "c=s" => \$comunidad,  "comunidad=s" => \$comunidad,
    );
}
$paquetes_perdidos;
$paquetes_salientes;
@datos_T1;
procesar_argumentos();
$fichero = $ip.'.'.$interfaz.':paq'. '2';
if (-e $fichero)
{
    # Ejecucion normal, ya tenemos referencias anteriores
    # Paquetes descartados
    my $oid = '.1.3.6.1.2.1.2.2.1.19.'.$interfaz;
    my $host = $ip;
    my $community = $comunidad;
    my ($session, $error) = Net::SNMP->session(-hostname =>
$host, -community => $community);
    my $result = $session->get_request(-varbindlist =>
[$oid],);
    if (!defined $result){
        printf "ERROR: %s.\n", $session->error();
        $session->close();
        exit 1;
    }
}

```

```

    my $paquetes_descartados = $result->{$oid};
    # Paquetes erroneos
    my $oid = '1.3.6.1.2.1.2.2.1.20.'.$interfaz;
    my $result = $session->get_request(-varbindlist =>
[$oid],);
    if (!defined $result){
        printf "ERROR: %s.\n", $session->error();
        $session->close();
        exit 1;
    }
    my $paquetes_erroneos = $result->{$oid};
    $paquetes_perdidos = $paquetes_descartados +
$paquetes_erroneos;
    # Paquetes salientes unicast
    my $oid = '.1.3.6.1.2.1.2.2.1.17.'.$interfaz;
    my $result = $session->get_request(-varbindlist =>
[$oid],);
    if (!defined $result){
        printf "ERROR: %s.\n", $session->error();
        $session->close();
        exit 1;
    }
    my $paquetes_unicast = $result->{$oid};
    # Paquetes salientes no-unicast
    my $oid = '.1.3.6.1.2.1.2.2.1.18.'.$interfaz;
    my $result = $session->get_request(-varbindlist =>
[$oid],);
    if (!defined $result){
        printf "ERROR: %s.\n", $session->error();
        $session->close();
        exit 1;
    }
    my $paquetes_no_unicast = $result->{$oid};
    $session->close();
    $paquetes_salientes = $paquetes_unicast +
$paquetes_no_unicast;
    open (fichero, $fichero) || die "no se puede abrir el
fichero.\n";
    my $line = <fichero>;
    close fichero;
    print @line;
    @datos_T1 = split(/;/, $line);
    }
    else
    {
        # Es la primera ejecucion
        # Paquetes descartados
        my $oid = '.1.3.6.1.2.1.2.2.1.19.'.$interfaz;
        my $host = $ip;
        my $community = $comunidad;
        my ($session, $error) = Net::SNMP->session(-hostname =>
$host, -community => $community);
        my $result = $session->get_request(-varbindlist =>
[$oid],);
        if (!defined $result){
            printf "ERROR: %s.\n", $session->error();
            $session->close();

```

```

        exit 1;
    }
    my $paquetes_descartados = $result->{$oid};
    # Paquetes erroneos
    my $oid = '.1.3.6.1.2.1.2.2.1.20.'. $interfaz;
    my $result = $session->get_request(-varbindlist =>
[$oid],);
    if (!defined $result){
        printf "ERROR: %s.\n", $session->error();
        $session->close();
        exit 1;
    }
    my $paquetes_erroneos = $result->{$oid};
    $paquetes_perdidos = $paquetes_descartados +
$paquetes_erroneos;
    # Paquetes salientes unicast
    my $oid = '.1.3.6.1.2.1.2.2.1.17.'. $interfaz;
    my $result = $session->get_request(-varbindlist =>
[$oid],);
    if (!defined $result){
        printf "ERROR: %s.\n", $session->error();
        $session->close();
        exit 1;
    }
    my $paquetes_unicast = $result->{$oid};
    # Paquetes salientes no-unicast
    my $oid = '.1.3.6.1.2.1.2.2.1.18.'. $interfaz;
    my $result = $session->get_request(-varbindlist =>
[$oid],);
    if (!defined $result){
        printf "ERROR: %s.\n", $session->error();
        $session->close();
        exit 1;
    }
    my $paquetes_no_unicast = $result->{$oid};
    $paquetes_salientes = $paquetes_unicast +
$paquetes_no_unicast;
    $session->close();
    open(fichero, '>'. $fichero) || die "no se puede abrir el
fichero\n";
    push (@datos, $paquetes_perdidos);
    push (@datos, $paquetes_salientes);
    my @registro = join (";", @datos);
    print fichero "@registro\n";
    close fichero;
    print 0;
    exit 0;
}
open (fichero, '>'. $fichero) || die "no se puede abrir el
fichero.\n";
push (@datos, $paquetes_perdidos);
push (@datos, $paquetes_salientes);
my @registro = join (";", @datos);
print fichero "@registro\n";
close fichero;
# Calculamos el porcentaje de perdidas
$incremento_perdidos = $paquetes_perdidos - $datos_T1[0];

```

```

    $incremento_entrantes =
$paquetes_salientes - $datos_T1[1];
    #Evitar divisiones por cero
    if ($incremento_entrantes != 0) {
        $perdidos = $incremento_perdidos /
$incremento_entrantes;
    } else {
        $perdidos = 0;
    }
    if ($perdidos < 0) { print 0; }
    else { print $perdidos; }

```

hirschmann_uso_poe.sh

```

#!/bin/sh

### Captura de argumentos

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$h" ] || [ -z "$c" ]; then
    echo "Uso: hirschmann_uso_poe.sh -h <Dir_IP_Switch> -c
<Comunidad_SNMP_v2>"
    exit 1
fi

### Consulta SNMP: PoE Máximo

PoE_Max=$(snmpget -v2c -c "$c" "$h"
.1.3.6.1.4.1.248.14.2.14.1.3.0 | cut -d " " -f4)

if [ "$PoE_Max" -eq 0 ]; then
    echo 100
    exit 0
fi

PoE_Max=$(( PoE_Max * 1000 ))

### Consulta SNMP: PoE 1, 2, 3, 4

PoE_1=$(snmpget -v2c -c "$c" "$h"
.1.3.6.1.4.1.248.14.2.14.2.1.2.1 | cut -d " " -f4)

PoE_2=$(snmpget -v2c -c "$c" "$h"
.1.3.6.1.4.1.248.14.2.14.2.1.2.2 | cut -d " " -f4)

PoE_3=$(snmpget -v2c -c "$c" "$h"
.1.3.6.1.4.1.248.14.2.14.2.1.2.3 | cut -d " " -f4)

PoE_4=$(snmpget -v2c -c "$c" "$h"
.1.3.6.1.4.1.248.14.2.14.2.1.2.4 | cut -d " " -f4)

### Cálculo de sumatorio

```

```
PoE_Sum=$(( PoE_1 + PoE_2 + PoE_3 + PoE_4 ))
### Cálculo de % de uso
echo $(( $( PoE_Sum *100 ) / PoE_Max ))
```

A3.2.- Scripts Servidores HP ProLiant

hp_ipmi_chassis_power.sh

```
#!/bin/sh

### Captura de argumentos

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo "hp_ipmi_chassis_power.sh -H <Dir_IP> -U
<Username> -P <Password>"
    exit 2
fi

### Consulta IPMI

result=$(ipmitool -H $H -U $U -P $P -L USER -I lanplus chassis
power status | cut -d " " -f4)

if [ "$result" == "on" ]; then
    echo 1
    exit 1
else
    echo 0
    exit 0
fi
```

hp_ipmi_power_meter.sh

```
#!/bin/sh

### Captura de argumentos

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo "hp_ipmi_power_meter.sh -H <Dir_IP> -U <Username> -P
<Password>"
    exit 2
fi

### Consulta IPMI
```

```
ipmitool -H $H -U $U -P $P -L USER -I lanplus sensor get
"Power Meter" | grep "Sensor Reading" | cut -d " " -f12
```

hp_ipmi_power_supply.sh

```
#!/bin/sh

### Captura de argumentos

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ] || [ -z "$N" ];
then
    echo "hp_ipmi_power_supply.sh -H <Dir_IP> -U
<Username> -P <Password> -N <Fuente_Aliment>"
    exit 2
fi
```

Consulta IPMI

```
ipmitool -H $H -U $U -P $P -L USER -I lanplus sensor get
"Power Supply $N" | grep "Sensor Reading" | cut -d " " -f12
```

hp_ipmi_temp_fuentes.sh

```
#!/bin/sh

### Captura de argumentos

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo "hp_ipmi_temp_fuentes.sh -H <Dir_IP> -U
<Username> -P <Password>"
    exit 2
fi

### Consulta IPMI y calculo de valor medio

ipmitool -H $H -I lanplus -U $U -P $P -L USER sdr get "Temp 8"
"Temp 9" | grep "Sensor Reading" | cut -d " " -f12 | awk
'BEGIN {t=0.0;c=0;a=0.0;} {t+=$1; c++;} END {printf("%f\n",
t/c);}'
```

hp_ipmi_temp_ram.sh

```
#!/bin/sh

### Captura de argumentos

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
```

```

    shift
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo "hp_ipmi_temp_ram.sh -H <Dir_IP> -U <Username> -P
<Password>"
    exit 2
fi

### Consulta IPMI y calculo de valor medio

ipmitool -H $H -I lanplus -U $U -P $P -L USER sdr get "Temp 4"
"Temp 5" "Temp 6" "Temp 7" | grep "Sensor Reading" | cut -d "
" -f12 | awk 'BEGIN {t=0.0;c=0;a=0.0;} {t+=$1; c++;} END
{printf("%f\n", t/c);}'

```

hp_ipmi_temp_ambiental.sh

```
#!/bin/sh
```

```
### Captura de argumentos
```

```
while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done
```

```
if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo "hp_ipmi_temp_ambiental.sh -H <Dir_IP> -U
<Username> -P <Password>"
    exit 2
fi
```

```
### Consulta IPMI y calculo de valor medio
```

```
ipmitool -H $H -I lanplus -U $U -P $P -L USER sdr get "Temp 1"
| grep "Sensor Reading" | cut -d " " -f12 | awk 'BEGIN
{t=0.0;c=0;a=0.0;} {t+=$1; c++;} END {printf("%f\n", t/c);}'

```

hp_ipmi_temp_disk.sh

```
#!/bin/sh
```

```
### Captura de argumentos
```

```
while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done
```

```
if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo "hp_ipmi_temp_disk.sh -H <Dir_IP> -U <Username> -P
<Password>"
    exit 2
fi
```

```
### Consulta IPMI y calculo de valor medio

ipmitool -H $H -I lanplus -U $U -P $P -L USER sdr get "Temp
29" | grep "Sensor Reading" | cut -d " " -f12 | awk 'BEGIN
{t=0.0;c=0;a=0.0;} {t+=$1; c++;} END {printf("%f\n", t/c);}'
```

hp_ipmi_fan.sh

```
#!/bin/sh
```

```
### Captura de argumentos
```

```
while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ] || [ -z "$N" ];
then
    echo "hp_ipmi_fan.sh -H <Dir_IP> -U <Username> -P
<Password> -N <ID_Ventilador>"
    exit 2
fi
```

```
### Consulta IPMI
```

```
ipmitool -H $H -U $U -P $P -L USER -I lanplus sensor get "Fan
$N" | grep "Sensor Reading" | cut -d " " -f12
```

check_cim_hpq.py (modificado)

***NOTA:** Este script es la base de el resto de scripts WBEM. Por comodidad, se ha creado un enlace dentro de `/usr/bin/` apuntando al script contenido en el directorio de `plug-ins` habitual `/usr/share/pandora_server/util/plugin/hp/`.

```
#!/usr/bin/env python
```

```
# First import the various python modules we'll need
import pywbem
import os
import sys
from optparse import OptionParser

# Some command line argument parsing gorp to make the script a
little more
# user friendly.
usage = '''Usage: %prog [options]

    This program will dump the version of ESX from a host
specified with the -s option.'''
parser = OptionParser(usage=usage)
parser.add_option('-s', '--server', dest='server',
help='Specify the server to connect to')
parser.add_option('-u', '--username', dest='username',
help='Username (default is root)')
parser.add_option('-p', '--password', dest='password',
help='Password (default is blank)')
```

```

parser.add_option('-c', '--cim', dest="cim", help='Objeto CIM
a consultar')
(options, args) = parser.parse_args()
if options.server is None:
    print 'You must specify a server to connect to. Use --help
for usage'
    sys.exit(1)
if options.cim is None:
    print 'Debes especificar un objeto CIM. Utiliza --help'
    sys.exit(1)
if options.username is None:
    options.username = 'root'
if options.password is None:
    options.password = ''

# Set up the client connection object.
# Hint -- With CIM XML, there is no concept of statefull
sessions
#         so this call doesn't actually connect to the server.
That
#         happens later once we send a request
client = pywbem.WBEMConnection('https://'+options.server,
(options.username, options.password), 'root/hpq')

# Now send an "Enumerate Instances" request. This will fetch
all
# instances of the specified class (or subclasses)
list = client.EnumerateInstanceNames(options.cim)
if list is None:
    print 'Error: No se encontraron instancias de '+options.cim
else:
    for n in list:
        resultado = client.GetInstance(n)
        for clave, valor in resultado.items():
            print '%s = %s' % (clave, valor)

```

check_esxcli_hpparray (modificado)

```

#!/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/
bin
PROGRAMNAME=`basename $0`
PROGPATH=`echo $0 | sed -e 's,[\\\/][^\\\/][^\\\/]*$,,'`
REVISION=v0.9
ESXCLI=/usr/bin/esxcli

STATE_OK=0
STATE_WARNING=1
STATE_CRITICAL=2
STATE_UNKNOWN=3
STATE_DEPENDENT=4

print_usage() {
    echo ""
    echo "Usage: $PROGRAMNAME -h <host> -u <username> -p
<password> -s <slot-number>"
    echo "Usage: $PROGRAMNAME [--help]"
    echo "Usage: $PROGRAMNAME [-V | --version]"
}

```

```

        echo ""
    }

print_help() {
    print_revision $PROGNAME $REVISION
    echo ""
    print_usage
    echo ""
    echo "This plugin checks hardware status for HP
Proliant running ESXi 5.0u1 servers using ESXCLI utility."
    echo ""
    exit 0
}

print_revision() {
    echo $1" "$2
}

if [ $# -lt 1 ]; then
    print_usage
    exit $STATE_UNKNOWN
fi

check_raid()
{
    raid_ok=`echo $check|grep -i ok|wc -l`
    raid_warning=`echo $check|grep -i rebuild|wc -l`
    raid_critical_1=`echo $check|grep -i failed|wc -l`
    raid_critical_2=`echo $check|grep -i recovery|wc -l`

    err_check=`expr $raid_ok + $raid_warning +
$raid_critical_1 + $raid_critical_2`

    if [ $err_check -eq "0" ]; then
        checkm=`echo $check|sed -e '/^$/ d`
        echo "$PROGNAME Error. $checkm"
        exit 2
    fi

    if [ $raid_critical_1 -ge "1" ]; then
        exit_status=$STATE_CRITICAL
    elif [ $raid_critical_2 -ge "1" ]; then
        exit_status=$STATE_CRITICAL
    elif [ $raid_warning -ge "1" ]; then
        exit_status=$STATE_WARNING
    elif [ $raid_ok -eq "0" ]; then
        exit_status=$STATE_UNKNOWN
    else
        exit_status=$STATE_OK
    fi

    if [ $exit_status -eq "0" ]; then
        msg_ok=`echo $check|grep -i ok`
        echo "RAID OK - ($msg_ok)"
        exit $exit_status
    elif [ $exit_status -eq "1" ]; then
        msg_warning=`echo $check|grep -i rebuild`

```

```

        echo "RAID WARNING - ($msg_warning)"
        exit $exit_status
    elif [ $exit_status -eq "2" ]; then
        msg_critical1=`echo $check|grep -i failed`
        msg_critical2=`echo $check|grep -i recovery`
        echo "RAID CRITICAL - ($msg_critical1
$msg_critical2)"
        exit $exit_status
    fi
}

```

```
### Captura de argumentos
```

```

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

```

```

if [ -z "$s" ]; then
    s=0
fi

```

```

if [ -z "$h" ] || [ -z "$u" ] || [ -z "$p" ]; then
    echo "check_esxcli_hparrray -h <Dir_IP> -u
<Username> -p <Password> -s <Slot>"
    exit 2
fi

```

```
### Chequeo mediante ESXCLI
```

```

check=`$ESXCLI -s $h -u $u -p $p hpacucli cmd -q "controller
slot=$s ld all show"`
check_raid

```

```
hp\_wbem\_chasis\_estado.sh
```

```
#!/bin/sh
```

```
### Captura de argumentos
```

```

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

```

```

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo "hp_wbem_chasis_estado.sh -H <Dir_IP> -U
<Username> -P <Password>"
    exit 2
fi

```

```

port=":5989"
H=$H$port

```

```
### Consulta WBEM y conteo
```

```
hp_cim_hpq.py -s $H -u $U -p $P -c "HP_ComputerSystemChassis"
| grep "HealthState =" | cut -d " " -f3 | awk '{if ($1==20)
print "Major Failure"; else if ($1==10) print "Degraded"; else
if ($1==5) print "OK";}'
```

hp_wbem_nic_estado.sh

```
#!/bin/sh

### Captura de argumentos

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo "hp_wbem_nic_estado.sh -H <Dir_IP> -U <Username> -P
<Password>"
    exit 2
fi

port=":5989"
H=$H$port
```

Consulta WBEM y conteo

```
hp_cim_hpq.py -s $H -u $U -p $P -c "HP_EthernetPort" | grep
"HealthState =" | cut -d " " -f3 | awk 'BEGIN
{ok=0;deg=0;cri=0} {$1==5?ok++:$1==10?deg++:cri++} END {if
(cri>0) print "Critical"; else if (deg>0) print "Degraded";
else print "OK";}'
```

hp_wbem_ps_estado.sh

```
#!/bin/sh

### Captura de argumentos

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo "hp_wbem_ps_estado.sh -H <Dir_IP> -U <Username> -P
<Password>"
    exit 2
fi

port=":5989"
H=$H$port
```

Consulta WBEM y conteo

```
hp_cim_hpq.py -s $H -u $U -p $P -c "SMX_PowerSupply" | grep
"HealthState =" | cut -d " " -f3 | awk 'BEGIN {ok=0;deg=0}
```

```
{$1==5?ok++:deg++} END {if (deg>0) print "Degraded"; else
print "OK";}'
```

hp_wbem_ram_cap.sh

```
#!/bin/sh
```

```
### Captura de argumentos
```

```
while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done
```

```
if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo "hp_wbem_ram_cap.sh -H <Dir_IP> -U <Username> -P
<Password>"
    exit 2
fi
```

```
port=":5989"
H=$H$port
```

```
### Consulta WBEM y sumatorio
```

```
hp_cim_hpq.py -s $H -u $U -p $P -c "CIM_PhysicalMemory" | grep
"Capacity =" | cut -d " " -f3 | awk 'BEGIN {t=0.0} {t+=$1} END
{printf("%f\n", t/2^30);}'
```

hp_wbem_ram_estado.sh

```
#!/bin/sh
```

```
### Captura de argumentos
```

```
while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done
```

```
if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo "hp_wbem_ram_estado.sh -H <Dir_IP> -U <Username> -P
<Password>"
    exit 2
fi
```

```
port=":5989"
H=$H$port
```

```
### Consulta WBEM y conteo
```

```
hp_cim_hpq.py -s $H -u $U -p $P -c "CIM_PhysicalMemory" | grep
"HealthState =" | cut -d " " -f3 | awk 'BEGIN
{ok=0;deg=0;cri=0} {$1==5?ok++:$1==10?deg++:cri++} END {if
(cri>0) print "Critical"; else if (deg>0) print "Degraded";
else print "OK";}'
```

hp_wbem_cpu_estado.sh

```
#!/bin/sh

### Captura de argumentos

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo "hp_wbem_cpu_estado.sh -H <Dir_IP> -U <Username> -P
<Password>"
    exit 2
fi

port=":5989"
H=$H$port

### Consulta WBEM y conteo

hp_cim_hpq.py -s $H -u $U -p $P -c "CIM_Processor" | grep
"HealthState =" | cut -d " " -f3 | awk 'BEGIN
{ok=0;deg=0;cri=0} {$1==5?ok++:$1==10?deg++:cri++} END {if
(cri>0) print "Critical"; else if (deg>0) print "Degraded";
else print "OK";}'
```

hp_wbem_ram_slots.sh

```
#!/bin/sh

### Captura de argumentos

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo "hp_wbem_ram_slots.sh -H <Dir_IP> -U <Username> -P
<Password>"
    exit 2
fi

port=":5989"
H=$H$port

### Consulta WBEM y conteo

hp_cim_hpq.py -s $H -u $U -p $P -c "CIM_PhysicalMemory" | grep
"Capacity =" | cut -d " " -f3 | awk 'BEGIN {c=0} {c++;} END
{printf("%f\n", c);}'
```

A3.3.- Scripts EMC² VNX/VNXe

emc_uemcli_ssd_estado.sh

```
#!/bin/sh

### Captura de argumentos

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo "emc_uemcli_ssd_estado.sh -H <Dir_IP> -U
<Username> -P <Password>"
    exit 2
fi

### Consulta CLI y conteo

uemcli -d $H -u $U -p $P -silent -sslPolicy accept /env/ssid
show | grep "Health state" | cut -d "(" -f2 | cut -d ")" -f1 |
awk 'BEGIN {max=0} {if ($1==0) $1=16; if ($1>max) max=$1;} END
{if (max==5) print "OK"; if (max==7) print "WARNING: O.K.
BUT"; if (max==10) print "WARNING: Degraded/Warning"; if
(max==15) print "WARNING: Minor failure"; if (max==20) print
"CRITICAL: Major failure"; if (max==25) print "CRITICAL:
Critical failure"; if (max==30) print "CRITICAL:
Non-recoverable error"; if (max==16) print "WARNING: Unknown";
}'
```

emc_uemcli_alertas.sh

```
#!/bin/sh

### Captura de argumentos

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo "emc_uemcli_pool_info.sh -H <Dir_IP> -U
<Username> -P <Password> [-dias <Dias_Antiguedad>]"
    exit 2
fi

if [ -z "$dias" ]; then
    dias=0;
fi

### Consulta CLI (Suprimimos alertas "notice" e "info")

NUM=0
CRITICAL=0
```

```

mkfifo pipe_while
uemcli -d $H -u $U -p $P -silent -sslPolicy accept
/event/alert/hist show -output csv | grep -v
'"Time","Message","Description","Severity"' | grep -v
',"notice"' | grep -v ', "info"' > pipe_while&
while read FILA
do
    ### Filtrado por antigüedad

    DIAS_FILA=$(( (($ (date -d $(date +%Y-%m-%d)
+%s) - $(date -d "$(echo $FILA | cut -d " " -f1 | sed
s/'"/"/g)" +%s) ) ) / 86400 )
    if [ $dias -ge $DIAS_FILA ]
    then
        echo $FILA | sed s/'"/"/g
        NUM=$(( $NUM + 1 ) )
        if [ $(echo $FILA | cut -d "," -f4) == "critical" ];
then
            CRITICAL=1
        fi
    fi
done < pipe_while

rm pipe_while

if [ "$CRITICAL" -gt 0 ]
then
    echo "____"
    echo ""
    echo "$NUM alertas mostradas, estado CRITICAL"
elif [ "$NUM" -gt 0 ]
then
    echo "____"
    echo ""
    echo "$NUM alertas mostradas, estado WARNING"
else
    echo "No hay alertas recientes, estado OK"
fi

emc_uemcli_bat.sh
#!/bin/sh

### Captura de argumentos

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo "emc_uemcli_bat_estado.sh -H <Dir_IP> -U
<Username> -P <Password>"
    exit 2
fi

```

```
### Consulta CLI y conteo
```

```
uemcli -d $H -u $U -p $P -silent -sslPolicy accept /env/bat
show | grep "Health state =" | cut -d " " -f11 | cut -c2 | awk
'BEGIN {max=0} {if ($1==0) $1=16; if ($1>max) max=$1;} END {if
(max==5) print "OK"; if (max==7) print "WARNING: O.K. BUT"; if
(max==10) print "WARNING: Degraded/Warning"; if (max==15)
print "WARNING: Minor failure"; if (max==20) print "CRITICAL:
Major failure"; if (max==25) print "CRITICAL: Critical
failure"; if (max==30) print "CRITICAL: Non-recoverable
error"; if (max==16) print "WARNING: Unknown"; }'
```

[emc_uemcli_discos_estado.sh](#)

```
#!/bin/sh
```

```
### Captura de argumentos
```

```
while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo "emc_uemcli_discos_estado.sh -H <Dir_IP> -U
<Username> -P <Password>"
    exit 2
fi
```

```
### Consulta CLI y conteo
```

```
uemcli -d $H -u $U -p $P -silent -sslPolicy accept /env/disk
show | grep "Health state" | cut -d "(" -f2 | cut -d ")" -f1 |
awk 'BEGIN {max=0} {if ($1==0) $1=16; if ($1>max) max=$1;} END
{if (max==5) print "OK"; if (max==7) print "WARNING: O.K.
BUT"; if (max==10) print "WARNING: Degraded/Warning"; if
(max==15) print "WARNING: Minor failure"; if (max==20) print
"CRITICAL: Major failure"; if (max==25) print "CRITICAL:
Critical failure"; if (max==30) print "CRITICAL:
Non-recoverable error"; if (max==16) print "WARNING: Unknown";
}'
```

[emc_uemcli_mm_estado.sh](#)

```
#!/bin/sh
```

```
### Captura de argumentos
```

```
while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo "emc_uemcli_mm_estado.sh -H <Dir_IP> -U
<Username> -P <Password>"
    exit 2
fi
```

```
fi
```

```
### Consulta CLI y conteo
```

```
uemcli -d $H -u $U -p $P -silent -sslPolicy accept /env/mm
show | grep "Health state" | cut -d "(" -f2 | cut -d ")" -f1 |
awk 'BEGIN {max=0} {if ($1==0) $1=16; if ($1>max) max=$1;} END
{if (max==5) print "OK"; if (max==7) print "WARNING: O.K.
BUT"; if (max==10) print "WARNING: Degraded/Warning"; if
(max==15) print "WARNING: Minor failure"; if (max==20) print
"CRITICAL: Major failure"; if (max==25) print "CRITICAL:
Critical failure"; if (max==30) print "CRITICAL:
Non-recoverable error"; if (max==16) print "WARNING: Unknown";
}'
```

```
emc\_uemcli\_nas\_info.sh
```

```
#!/bin/sh
```

```
### Captura de argumentos
```

```
while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done
```

```
if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ] || [ -z "$ID" ];
then
    echo "emc_uemcli_nas_info.sh -H <Dir_IP> -U <Username> -P
<Password> -ID <ID_Shared_Folder>"
    exit 2
fi
```

```
### Consulta CLI
```

```
uemcli -d $H -u $U -p $P -silent -sslPolicy accept
/stor/config/sf/res -id $ID show
```

```
emc\_uemcli\_nas\_uso.sh
```

```
#!/bin/sh
```

```
### Captura de argumentos
```

```
while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done
```

```
if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ] || [ -z "$ID" ];
then
    echo "emc_uemcli_nas_uso.sh -H <Dir_IP> -U <Username> -P
<Password> -ID <ID_Shared_Folder>"
    exit 2
fi
```

```
### Consulta CLI
```

```
QUERY=$(uemcli -d $H -u $U -p $P -silent -sslPolicy accept
/stor/config/sf/res -id $ID show -output csv)
```

```
### Calculo de porcentaje
```

```
TOTAL=$(echo $QUERY | cut -d "," -f8 | cut -d " " -f1)
```

```
USADA=$(echo $QUERY | cut -d "," -f9 | cut -d " " -f1)
```

```
echo "$USADA $TOTAL" | awk '{printf "%f\n", $1*100/$2}'
```

[emc_uemcli_pool_info.sh](#)

```
#!/bin/sh
```

```
### Captura de argumentos
```

```
while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done
```

```
if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ] || [ -z "$pool"
]; then
```

```
    echo "emc_uemcli_pool_info.sh -H <Dir_IP> -U
<Username> -P <Password> -pool <Nombre_Pool>"
    exit 2
fi
```

```
### Consulta CLI
```

```
ID=$(uemcli -d $H -u $U -p $P -silent -sslPolicy accept
/stor/config/pool show -output csv | grep $pool | sed s/'"/'//g
| cut -d "," -f1)
```

```
QUERY=$(uemcli -d $H -u $U -p $P -silent -sslPolicy accept
/stor/config/pool -id $ID show)
```

```
echo $QUERY
```

```
if [ -z "$QUERY" ]; then
    echo "CRITICAL: Pool no encontrado"
fi
```

[emc_uemcli_pool_uso.sh](#)

```
#!/bin/sh
```

```
### Captura de argumentos
```

```
while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done
```

```

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ] || [ -z "$pool"
]; then
    echo "emc_uemcli_pool_uso.sh -H <Dir_IP> -U <Username> -P
<Password> -pool <Nombre_Pool>"
    exit 2
fi

```

```

### Consulta CLI

```

```

uemcli -d $H -u $U -p $P -silent -sslPolicy accept
/stor/config/pool show -output csv | grep $pool | cut -d
", " -f5 | sed s/' ' //g | sed s/'%' //g

```

emc_uemcli_pools_estado.sh

```

#!/bin/sh

```

```

### Captura de argumentos

```

```

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

```

```

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo "emc_uemcli_pools_estado.sh -H <Dir_IP> -U
<Username> -P <Password>"
    exit 2
fi

```

```

### Consulta CLI y conteo

```

```

uemcli -d $H -u $U -p $P -silent -sslPolicy accept
/stor/config/pool show | grep "Health state" | cut -d "(" -f2
| cut -d ")" -f1 | awk 'BEGIN {max=0} {if ($1==0) $1=16; if
($1>max) max=$1;} END {if (max==5) print "OK"; if (max==7)
print "WARNING: O.K. BUT"; if (max==10) print "WARNING:
Degraded/Warning"; if (max==15) print "WARNING: Minor
failure"; if (max==20) print "CRITICAL: Major failure"; if
(max==25) print "CRITICAL: Critical failure"; if (max==30)
print "CRITICAL: Non-recoverable error"; if (max==16) print
"WARNING: Unknown"; }'

```

emc_uemcli_port_estado.sh

```

#!/bin/sh

```

```

### Captura de argumentos

```

```

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

```

```

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ] || [ -z "$port"
]; then

```

```

    echo "emc_uemcli_port_estado.sh -H <Dir_IP> -U
<Username> -P <Password> -port <Puerto>"
    exit 2
fi

### Consulta CLI y conteo

QUERY=$(uemcli -d $H -u $U -p $P -silent -sslPolicy accept
/net/port show -output csv | grep "$port" | cut -d "(" -f2 |
cut -d ")" -f1 | awk 'BEGIN {max=0} {if ($1==0) $1=16; if
($1>max) max=$1;} END {if (max==5) print "OK"; if (max==7)
print "WARNING: O.K. BUT"; if (max==10) print "WARNING:
Degraded/Warning"; if (max==15) print "WARNING: Minor
failure"; if (max==20) print "CRITICAL: Major failure"; if
(max==25) print "CRITICAL: Critical failure"; if (max==30)
print "CRITICAL: Non-recoverable error"; if (max==16) print
"WARNING: Unknown"; }')

echo $QUERY

if [ -z "$QUERY" ]; then
    echo "CRITICAL: Puerto(s) no encontrado(s)"
fi

```

[emc_uemcli_ps_estado.sh](#)

```
#!/bin/sh
```

```
### Captura de argumentos
```

```
while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
done
```

```
if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo "emc_uemcli_ps_estado.sh -H <Dir_IP> -U
<Username> -P <Password>"
    exit 2
fi
```

```
### Consulta CLI y conteo
```

```
uemcli -d $H -u $U -p $P -silent -sslPolicy accept /env/ps
show | grep "Health state =" | cut -d " " -f11 | cut -c2 | awk
'BEGIN {max=0} {if ($1==0) $1=16; if ($1>max) max=$1;} END {if
(max==5) print "OK"; if (max==7) print "WARNING: O.K. BUT"; if
(max==10) print "WARNING: Degraded/Warning"; if (max==15)
print "WARNING: Minor failure"; if (max==20) print "CRITICAL:
Major failure"; if (max==25) print "CRITICAL: Critical
failure"; if (max==30) print "CRITICAL: Non-recoverable
error"; if (max==16) print "WARNING: Unknown"; }'
```

[emc_uemcli_rep_info.sh](#)

```
#!/bin/sh
```

```
### Captura de argumentos
```

```

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ] || [ -z "$ID" ];
then
    echo "emc_uemcli_rep_info.sh -H <Dir_IP> -U <Username> -P
<Password> -ID <ID_Rep_Session>"
    exit 2
fi

### Consulta CLI

QUERY=$(uemcli -d $H -u $U -p $P -silent -sslPolicy accept
/prot/rep/session -id $ID show)

echo $QUERY

if [ -z "$QUERY" ]; then
    echo "CRITICAL: No encontrada"
fi

emc_uemcli_snaps_info.sh
#! /bin/sh

### Captura de argumentos

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo "emc_uemcli_discos_estado.sh -H <Dir_IP> -U
<Username> -P <Password>"
    exit 2
fi

### Consulta CLI

QUERY=$(uemcli -d $H -u $U -p $P -silent -sslPolicy accept
/prot/snap show)

echo $QUERY

if [ -z "$QUERY" ]; then
    echo "CRITICAL: No existen snapshots en el sistema"
fi

emc_uemcli_sp_estado.sh
#! /bin/sh

### Captura de argumentos

```

```

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo "emc_uemcli_sp_estado.sh -H <Dir_IP> -U
<Username> -P <Password>"
    exit 2
fi

### Consulta CLI y conteo

uemcli -d $H -u $U -p $P -silent -sslPolicy accept /env/sp
show | grep "Health state" | cut -d "(" -f2 | cut -d ")" -f1 |
awk 'BEGIN {max=0} {if ($1==0) $1=16; if ($1>max) max=$1;} END
{if (max==5) print "OK"; if (max==7) print "WARNING: O.K.
BUT"; if (max==10) print "WARNING: Degraded/Warning"; if
(max==15) print "WARNING: Minor failure"; if (max==20) print
"CRITICAL: Major failure"; if (max==25) print "CRITICAL:
Critical failure"; if (max==30) print "CRITICAL:
Non-recoverable error"; if (max==16) print "WARNING: Unknown";
}'

```

A3.4.- Scripts VMware ESXi

esxi_cpu_uso.sh

```
#!/bin/sh
```

```
### Captura de argumentos
```

```

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ]; then
    echo 'esxi_cpu_uso.sh -H <Dir_IP> -U <Username> -P
<Password>'
    exit 2
fi

```

```
### SSH: Comandos ESXi
```

```

sshpass -p $P ssh $U@$H <<'ENDSSH'
esxtop -b -import-entity "PCPU UTIL(%)" -n 1 -d 2 | cut -d
',' -f25 | grep -v "Physical Cpu" | sed 's/"//g'
ENDSSH

```

esxi_mem_uso.sh

```
#!/bin/sh
```

```
### Captura de argumentos
```

```

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$H" ] || [ -z "$C" ] || [ -z "$O" ]; then
    echo 'esxi_mem_uso.sh -H <Dir_IP> -C <Comunidad> -O
<"RAM" | "Disco">'
    exit 2
fi

### Descubrimiento de Index en hrStorageTable

if [ $O == "RAM" ]; then
    SUFIJO_OID=$(snmpwalk -v2c -c $C -OeU $H
.1.3.6.1.2.1.25.2.3.1.3 | grep "Real Memory" | cut -d " " -f1
| cut -d "." -f2)
elif [ $O == "Disco" ]; then
    SUFIJO_OID=$(snmpwalk -v2c -c $C -OeU $H
.1.3.6.1.2.1.25.2.3.1.2 | grep "hrStorageFixedDisk" | cut -d "
" -f1 | cut -d "." -f2)
else
    echo 'esxi_mem_uso.sh -H <Dir_IP> -C <Comunidad> -O
<"RAM" | "Disco">'
    exit 2
fi

### Consulta SNMP

TOTAL=$(snmpwalk -v2c -c $C -OeU $H
.1.3.6.1.2.1.25.2.3.1.5.$SUFUJO_OID | cut -d " " -f4)

USADA=$(snmpwalk -v2c -c $C -OeU $H
.1.3.6.1.2.1.25.2.3.1.6.$SUFUJO_OID | cut -d " " -f4)

echo "$USADA $TOTAL" | awk '{printf "%f\n", $1*100/$2}'

esxi_ram_allocation.sh
#!/bin/sh

### Captura de argumentos

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$H" ] || [ -z "$C" ]; then
    echo 'esxi_ram_allocation.sh -H <Dir_IP> -C <Comunidad>'
    exit 2
fi

### Consulta RAM Total

```

```
TOTAL=$(snmpwalk -v2c -c $C $H .1.3.6.1.4.1.6876.3.2.1.0 |
cut -d " " -f4)

### Sumatorio RAM reservada en VMs arrancadas

USADA=0
for i in $(snmpwalk -v2c -c $C $H .1.3.6.1.4.1.6876.2.1.1.6 |
grep "powered on" | cut -d " " -f1 | cut -d "." -f7)
do
    USADA=$((USADA+$(snmpget -v2c -c $C $H
.1.3.6.1.4.1.6876.2.1.1.5.$i | cut -d " " -f4)))
done

### Porcentaje

echo "$USADA $TOTAL" | awk '{printf "%f\n", $1*102400/$2}'
```

esxi_disk_iostats.sh

```
#!/bin/sh

### Captura de argumentos

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ] || [ -z "$valor"
]; then
    echo 'esxi_cpu_uso.sh
-H <Dir_IP> -U <Username> -P <Password>
-valor <KernelWrite | KernelRead | DriverWrite | DriverRead>
[-adapter <Disk_Adapter>] [-tempfile <Ruta_Fich_Temp>]'
    exit 2
fi

if [ -z "$adapter" ]; then
    adapter="vmhbal"
fi

if [ -z "$tempfile" ]; then
    tempfile="/tmp/esxtop_diskiostats_$adapter"
fi

case $valor in
    "KernelWrite" ) field=98;;
    "KernelRead" ) field=94;;
    "DriverWrite" ) field=97;;
    "DriverRead" ) field=93;;
    * ) echo "Valor $valor incorrecto"; exit 2
esac

### SSH: Comandos ESXi

sshpass -p $P ssh -T $U@$H <<ENDSSH
printf "%s\n%s\n" "Adapter" "$adapter" > $tempfile
```

```

        esxtop -b -n 3 -d 2 -a -import-entity "$tempfile" |
cut -d ',' -f$field | tail -n1 | sed s/' '//g
ENDSSH

```

A3.5.- Scripts Oracle Database Server

oracle_sqlplus.sh

```

#!/bin/sh

### Library Path: MODIFICAR O ELIMINAR

LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/lib/oracle/11.2/client/1
ib/

which sqlplus > /dev/null

### Captura de argumentos

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ] || [ -z "$SID" ]
|| [ -z "$PARAM" ]; then
    echo "oracle_sqlplus.sh -H <Servidor> -U <Username> -P
<Password> -SID <SID> [-port <Puerto>] -PARAM <ErroresORA |
Sesiones | Procesos | Transacciones>"
    exit 2
fi

if [ -z "$port" ]; then
    port=1521
fi

### SQL*Plus Connection

ErroresORA () {
    QUERY=$(sqlplus -S
"$U/$P@(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST
=$H)(PORT=$port)))(CONNECT_DATA=(SID=$SID)))" << EOF

        SET HEADING OFF
        SELECT NAME, TEXT FROM SYS.ALL_ERRORS WHERE TEXT LIKE
'%ORA-%';
        exit
    EOF)
    if [ -z "$QUERY" ]; then
        echo "OK: No hay errores ORA-XXXX"
    else
        echo "CRITICAL: Se han encontrado los siguientes
errores:"
        echo
        "-----"
        echo ""
    fi
}

```

```

        echo "$QUERY"
    fi
}

Sesiones () {
    QUERY=$(sqlplus -S
"$U/$P@(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST
=$H)(PORT=$port)))(CONNECT_DATA=(SID=$SID)))" << EOF

        SET HEADING OFF
        SELECT COUNT(SID) FROM V\$$SESSION;
        SELECT VALUE FROM V\$$PARAMETER WHERE NAME='sessions';
        exit

    EOF)
    SESIONES_ACTUALES=$(echo $QUERY | cut -d " " -f1)
    SESIONES_MAX=$(echo $QUERY | cut -d " " -f2)
    echo "$SESIONES_ACTUALES $SESIONES_MAX" | awk '{printf "%f\n",
$1*100/$2}'
}

Procesos () {
    QUERY=$(sqlplus -S
"$U/$P@(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST
=$H)(PORT=$port)))(CONNECT_DATA=(SID=$SID)))" << EOF

        SET HEADING OFF
        SELECT COUNT(PID) FROM V\$$PROCESS;
        SELECT VALUE FROM V\$$PARAMETER WHERE NAME='processes';
        exit

    EOF)
    PROCESOS_ACTUALES=$(echo $QUERY | cut -d " " -f1)
    PROCESOS_MAX=$(echo $QUERY | cut -d " " -f2)
    echo "$PROCESOS_ACTUALES $PROCESOS_MAX" | awk '{printf "%f\n",
$1*100/$2}'
}

Transacciones () {
    QUERY=$(sqlplus -S
"$U/$P@(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST
=$H)(PORT=$port)))(CONNECT_DATA=(SID=$SID)))" << EOF

        SET HEADING OFF
        SELECT COUNT(XID) FROM V\$$TRANSACTION;
        SELECT VALUE FROM V\$$PARAMETER WHERE
NAME='transactions';
        exit

    EOF)
    TRANS_ACTUALES=$(echo $QUERY | cut -d " " -f1)
    TRANS_MAX=$(echo $QUERY | cut -d " " -f2)
    echo "$TRANS_ACTUALES $TRANS_MAX" | awk '{printf "%f\n",
$1*100/$2}'
}

case $PARAM in

```

```

    "ErroresORA")
        ErroresORA;;
    "Sesiones")
        Sesiones;;
    "Procesos")
        Procesos;;
    "Transacciones")
        Transacciones;;
*)
    echo "El argumento -PARAM $PARAM no es correcto"
    echo "Posibles valores:"
    echo " -PARAM <ErroresORA | Sesiones | Procesos |
Transacciones>;";
esac

```

oracle_db_estado.sh

```

#!/usr/bin/perl

$ENV{"LD_LIBRARY_PATH"} = '/usr/lib/oracle/11.2/client/lib/';

use Getopt::Long;

GetOptions('H=s' => \$host, 'port=i' => \$port, 'SID=s' =>
\$sid, 'U=s' => \$user, 'P=s' => \$pass, 'modo=i' => \$modo);

$errores = 0;

sub trim($);
my @result;
my @param_array = (
    [90,">","Dictionary Cache Hit Ratio",'SELECT
(1 - (Sum(getmisses)/(Sum(gets) + Sum(getmisses)))) * 100 FROM
v\$rowcache;'],
    [99,">","Library Cache Hit Ratio",'SELECT
(1 - (Sum(reloads)/(Sum(pins) + Sum(reloads)))) * 100 FROM
v\$librarycache;'],
    [85,">","DB Block Buffer Cache Hit Ratio",'SELECT
(1 - (phys.value / (db.value + cons.value))) * 100 FROM
v\$sysstat phys,v\$sysstat db,v\$sysstat cons WHERE phys.name
= \'physical reads\' AND db.name = \'db block gets\' AND
cons.name = \'consistent gets\';'],
    [98,">","Latch Hit Ratio",'SELECT (1 - (Sum(misses) /
Sum(gets))) * 100 FROM v\$latch;'],
    [5,"<","Disk Sort Ratio",'SELECT (disk.value/mem.value) *
100 FROM v\$sysstat disk,v\$sysstat mem WHERE disk.name =
\'sorts (disk)\' AND mem.name = \'sorts (memory)\';'],
    [5,"<","Rollback Segment Waits",'SELECT (Sum(waits) /
Sum(gets)) * 100 FROM v\$rollstat;'],
    [50,"<","Dispatcher Workload",'SELECT NVL((Sum(busy) /
(Sum(busy) + Sum(idle))) * 100,0) FROM v\$dispatcher;']
);
# is possible define own selects [reference value,operator
(<;>;eq;ne etc.),Description,select]
my @results;

sub array_rows {
    my ($array_rows) = @_;

```

```

    my $rows = @$array_rows;
    return $rows;
}

sub trim($)
{
    my $string = shift;
    $string =~ s/^\s+//;
    $string =~ s/\s+$//;
    return $string;
}

sub logon {
open (SQL,"sqlplus -s
pandorafms/mismatch@\(\(DESCRIPTION=\(\(ADDRESS=\(\(PROTOCOL=TCP\
\)\)\(Host=$host\)\)\(Port=$port\)\)\)\)\(CONNECT_DATA=\(\(SID=$s
id\)\)\)\)\)</dev/null|" ) or die;
    while ( my $res = <SQL> )
        {
            if ($res =~ /^(ORA-\d{5})/) {return $1;}
        }
}

if (logon() eq "ORA-01017"){

    for (my $i=0; $i<array_rows(\@param_array); $i++){
        # print
"$param_array[$i][0] -- $param_array[$i][1] -- $param_array[$i
][2] -- $param_array[$i][3]\n";

open (SQL,"sqlplus -s
$user/$pass@\(\(DESCRIPTION=\(\(ADDRESS=\(\(PROTOCOL=TCP\)\)\(Hos
t=$host\)\)\(Port=$port\)\)\)\)\(CONNECT_DATA=\(\(SID=$sid\)\)\)
\)\) << EOF
set pagesize 0
set numformat 999.999
$param_array[$i][3]
EOF |" ) or die;
    while ( my $res = <SQL> )
        {
            if ( $res =~ /\s*\S+/ ) { push(@results,trim($res));}
        }

    #print "$results[$modo]". "\n";
    printf("%.3f\n", $results[$modo]);

} else {print "CRITICAL: Imposible conectarse a Oracle (SID
$sid)\n";}

```

oracle_listener.sh

```
#!/bin/sh
```

```
### Variables de entorno: MODIFICAR O ELIMINAR
```

```

LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/lib/oracle/11.2/client/1
ib/
export LD_LIBRARY_PATH

ORACLE_HOME=/usr/share/pandora_server/app/pandora/product/11.2
.0/client_4/
export ORACLE_HOME

PATH=$PATH:/usr/local/bin
export PATH

which tnspring >& /dev/null
if [ $? -ne 0 ]; then
    echo "CRITICAL: No se ha encontrado el comando TNSPING"
    exit 2
fi

### Captura de argumentos

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$H" ] ; then
    echo "oracle_listener.sh -H <Servidor> [-port
<Puerto>]"
    exit 2
fi

if [ -z "$port" ]; then
    port=1521
fi

tnspring "(ADDRESS=(PROTOCOL=TCP)(Host=$H)(Port=$port))" 3
>&/dev/null && echo "OK: El listener responde correctamente"
|| echo "CRITICAL: El listener no responde correctamente"

exit 0

```

A3.6.- Scripts Tomcat Server

tomcat_info.sh

```

#! /bin/sh

### Captura de argumentos

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ] || [ -z "$PARAM"
]; then

```

```

        echo "$0 -H <Servidor> -U <Username> -P <Password> [-port
<Puerto>] -PARAM <Uso_Memoria | Uso_Threads | Peticiones |
Errores | Bytes_Subida | Bytes_Bajada>"
        exit 2
    fi

    if [ -z "$port" ]; then
        port=8080
    fi

    FICH="/tmp/tomcat_${H}:${port}.xml"

    ### Descargar $H/manager/status?XML=true si no existe o si no
    es reciente

    Descargar_XML () {
        FICHERO_ACTUALIZADO=$(find $FICH -mmin -5 2> /dev/null)
        if [ -z $FICHERO_ACTUALIZADO ] ; then

            wget -q --http-user=$U --http-password=$P --output-docume
nt=$FICH http://$H:$port/manager/status?XML=true
            fi
        }

    ### Uso de memoria del JVM (Total / Max)

    Uso_Memoria () {
        DATOS=$(cat $FICH | sed s/\>\</"\n"/g | grep 'memory
free=')
        USO=$(echo $DATOS | cut -d '"' -f4)
        MAX=$(echo $DATOS | cut -d '"' -f6)
        echo "$USO $MAX" | awk '{printf "%f\n", $1*100/$2}'
    }

    ### Uso de threads (currentThreadCount / maxThreads)

    Uso_Threads () {
        DATOS=$(cat $FICH | sed s/connector/"\n"/g | grep
"name=' ' | grep "\-$port" | sed s/\>\</"\n"/g)
        DATOS=$(echo "$DATOS" | grep "threadInfo")
        USO=$(echo $DATOS | cut -d '"' -f4)
        MAX=$(echo $DATOS | cut -d '"' -f2)
        echo "$USO $MAX" | awk '{printf "%f\n", $1*100/$2}'
    }

    ### Peticiones HTTP/HTTPS

    Peticiones () {
        DATOS=$(cat $FICH | sed s/connector/"\n"/g | grep
"name=' ' | grep "\-$port" | sed s/\>\</"\n"/g)
        echo "$DATOS" | grep "requestInfo" | cut -d '"' -f6
    }

    ### Errores en peticiones HTTP/HTTPS

    Errores () {

```

```

        DATOS=$(cat $FICH | sed s/connector/"\n"/g | grep
"name=" | grep "\-$port" | sed s/\>\</"\n"/g)
        echo "$DATOS" | grep "requestInfo" | cut -d '"' -f8
    }

### Bytes de subida

Bytes_Subida () {
    DATOS=$(cat $FICH | sed s/connector/"\n"/g | grep
"name=" | grep "\-$port" | sed s/\>\</"\n"/g)
    echo "$DATOS" | grep "requestInfo" | cut -d '"' -f12
}

### Bytes de bajada

Bytes_Bajada () {
    DATOS=$(cat $FICH | sed s/connector/"\n"/g | grep
"name=" | grep "\-$port" | sed s/\>\</"\n"/g)
    echo "$DATOS" | grep "requestInfo" | cut -d '"' -f10
}

### Tiempo total de procesamiento, en ms

Tiempo_Procesamiento () {
    DATOS=$(cat $FICH | sed s/connector/"\n"/g | grep
"name=" | grep "\-$port" | sed s/\>\</"\n"/g)
    echo "$DATOS" | grep "requestInfo" | cut -d '"' -f4
}

case $PARAM in
    "Uso_Memoria")
        Descargar_XML
        Uso_Memoria;;
    "Uso_Threads")
        Descargar_XML
        Uso_Threads;;
    "Peticiones")
        Descargar_XML
        Peticiones;;
    "Errores")
        Descargar_XML
        Errores;;
    "Bytes_Subida")
        Descargar_XML
        Bytes_Subida;;
    "Bytes_Bajada")
        Descargar_XML
        Bytes_Bajada;;
    "Tiempo_Procesamiento")
        Descargar_XML
        Tiempo_Procesamiento;;
    *)
        echo "El argumento -PARAM $PARAM no es correcto"
        echo "Posibles valores:"
        echo " -PARAM <Uso_Memoria | Uso_Threads |
Peticiones | Errores | Bytes_Subida | Bytes_Bajada>";;
esac

```

tomcat_app.sh

```
#!/bin/sh

### Captura de argumentos

while echo $1 | grep ^- > /dev/null; do
    eval $( echo $1 | sed 's/-//g' | tr -d '\012')=$2
    shift
    shift
done

if [ -z "$H" ] || [ -z "$U" ] || [ -z "$P" ] || [ -z "$app" ];
then
    echo "$0 -H <Servidor> -U <Username> -P <Password> [-port
<Puerto>] -app <Aplicacion>"
    exit 2
fi

if [ -z "$port" ]; then
    port=8080
fi

ESTADO=$(wget -q -O - --http-user=$U --http-password=$P
http://$H:$port/manager/list | grep "\/$app:running")
if [ -z "$ESTADO" ]; then
    echo "CRITICAL: App \"$app\" no lanzada"
    exit 2
fi
echo "OK: App \"$app\" lanzada"
exit 0
```