Monitorización con Agentes Software

Monitorización con Agentes Software

Monitorización con Agentes Software

Los Agentes Software se encuentran en ejecución en los sistemas operativos de los cuales recogen información, realizando un chequeo para cada módulo.

Las *directivas* propias del Agente Software sirven para recoger ciertos datos directamente del sistema operativo (ej. uso de CPU, memoria, eventos, etcétera), ejecutando comandos propios del sistema operativo siguiendo instrucciones de *scripts* predefinidos.

El Dataserver Pandora FMS procesa y almacena en la base de datos toda la información generada y enviada en fichero XML por los agentes software.

Configuración de Agentes Software

Toda la configuración y los parámetros se encuentran almacenados en el fichero *pandora_agent.conf*, el cual está instalado también localmente junto a su Agente Software. La configuración básica está tratada en "Configuración de los Agentes Pandora FMS", a continuación se expone la configuración avanzada.

Configuración local

En el archivo de configuración del Agente Software los módulos están definidos con la siguiente estructura básica de texto:

```
module_begin
module_name <your module name>
module_type generic_data
module_exec <your command>
module_description <your description>
module_end
```

- Para el Agente Software en MS Windows® y la instrucción `module_name`, si desea o necesita utilizar caracteres ASCII extendidos (áéíóú, por ejemplo) debe utilizar un *plugin* o *script externo.* Consulte la sección de plugin para Agentes Software.
- Para el Agente Software en MS Windows® se dispone, además, de `module_exec_powershell` para la ejecución nativa de chequeos con PowerShell®.

Configuración remota

Para habilitar la configuración remota se habilita el parámetro: `remote_config 1` y se reinicia el agente software.

Es posible gestionar remotamente los ficheros de los Agentes Software desde la Consola Web de Pandora FMS. La configuración de cada agente se almacena en el servidor de Pandora FMS en dos ficheros: <md5>.conf y <md5>.md5, donde <md5> es el *hash* del nombre del Agente software. Estos ficheros son almacenados respectivamente en:

```
/var/spool/pandora/data_in/conf
```

y

```
/var/spool/pandora/data_in/md5
```

Una vez que la configuración remota del agente está activada, cualquier cambio que se haga localmente en el fichero de configuración será sobreescrita por la configuración almacenada en la consola. Para volver a la administración local del Agente Software, detenga su servicio, restablezca `remote_config` a cero e inicie el servicio de nuevo.

Custom fields

Los campos personalizados permiten añadir información adicional al agente. Se pueden crear campos personalizados con la API 1.0 PFMS y el comando set create_custom_field o por medio de la Consola web en el menú Management → Resources → Custom fields → Create field.

- Las opciones Enabled combo, Password type y Link type son mutuamente excluyentes, es decir, solamente se podrá utilizar una de ellas (o ninguna, valor por defecto).
- Al activar el campo Display up front, la información del campo personalizado se mostrará, *en caso de tener algún valor establecido*, en la vista general del agente. Además, será necesario activar este *token* para enviar la información de los *Custom Fields* al Command Center (Metaconsola).
- Enabled combo: Este parámetro permite activar la configuración de parámetros seleccionables desde un desplegable. Una vez activado, en la ventana de configuración del custom field correspondiente aparecerá un nuevo campo para introducir los valores del combo separados por comas.
- Password type: El valor del campo (contraseña) será mostrado por medio de asteriscos en la Consola web.
- Link type: Permite agregar un campo personalizado que albergará un enlace web a rellenar por Consola web o en un XML recibido por un agente. Es posible incluir enlaces en los *custom fields* de un XML en formato JSON incrustados con instrucciones CDATA <![CDATA[…]]>. Por ejemplo, si el formato JSON del enlace es:

```
["Web name","https://example.com"]
```

El XML tendría esta sintaxis:

```
<custom_fields>
  <name>![CDATA[web]]</name>
  <value>![CDATA[["Web name","https://example.com"|]]]</value>
</custom_fields>
```

Consulte "Validación de XML", la Arquitectura de Seguridad para el protocolo Tentacle (mecanismo encargado de entregar datos en formato XML al Data server PFMS) y la Arquitectura de Seguridad para el Data server PFMS (limitar la autocreación de agentes y establecer una contraseña para el grupo de agentes al cual pertenezca cada agente).

Los *custom fields* también se pueden pasar desde el fichero de configuración del agente, utilizando los *token* custom_fieldx_name y custom_fieldx_value, por ejemplo:

```
custom_field1_name Serial Number
custom_field1_value 56446456KS7000
```

El campo personalizado llamado `Serial Number` viene creado por defecto al instalar PFMS y se pueden crear tantos campos personalizados como se necesiten y de cada tipo distinto (valor simple, enlace web, tipo contraseña y tipo lista de opciones). Es indiferente orden del identificador numérico de cada campo personalizado, solamente se debe asegurar que el nombre sea exactamente el mismo:

```
custom_field11_name Simple custom field name
custom_field11_value Simple custom field value

custom_field12_name Custom field Link type
custom_field12_value ["Pandora FMS web site","https://pandorafms.com"]

custom_field13_name Custom field Password type
custom_field13_value My;Password;

custom_field14_name Custom field Combo type
custom_field14_value Two
```

En los campos personalizados `Combo type` el valor enviado por al agente software debe corresponder exactamente con alguno de los item del mismo, de lo contrario el valor no será cambiado.

Parámetros de configuración comunes

Parámetros más importantes para la configuración básica de Agentes Software:

- server_ip: Dirección IP del servidor de Pandora FMS.
- server_path: Ruta de la carpeta de entrada *incoming* del servidor Pandora FMS, por defecto `/var/spool/pandora/data_in`.
- temporal: Carpeta, por defecto `/tmp`.
- logfile: Archivo de *log* del Agente Software, por defecto `/var/log/pandora/pandora_agent.log`.
- interval: Intervalo de ejecución del agente, por defecto `300` segundos.

Grupos protegidos por contraseña

Por defecto, cuando un agente envía datos por primera vez al servidor de Pandora FMS se añade de forma automática al grupo que se haya definido en el fichero de configuración del agente.

Es posible configurar una contraseña para un grupo, de esta manera un agente no se añadirá a un grupo a menos que se haya especificado la contraseña correcta en el fichero de configuración del agente.

Para editar y agregar una contraseña de grupo vaya al menú Management → Profiles → Manage agent groups → clic en nombre de grupo.

Para añadir un agente nuevo a este grupo, edite su fichero de configuración y añada la siguiente opción de configuración `group_password` y reinicie el agente software.

Módulos en Agentes y Agentes Software

Tipos de módulos

Según dato devuelto:

- generic_data: Numérico.
- generic_data_inc: Incremental.
- generic_data_inc_abs: Absoluto incremental.
- generic_proc: *Booleano* .
- generic_data_string: Alfanuméricos.
- async_data: Numérico asíncrono.
- async_string: Alfanumérico asíncrono.
- async_proc: Booleano asíncrono.
- Modulo de imagen: utilizan como base un módulo de tipo cadena de texto (`generic_data_string` o `async_string`). Si el dato que contiene el módulo es una imagen codificada en base64,(encabezado `data:image`) será identificado como una imagen y habilitará en las vistas un enlace a una ventana para recuperar la imagen. Además se mostrarán en su respectivo histórico un contenido de las distintas imágenes que conforman las cadenas almacenadas.

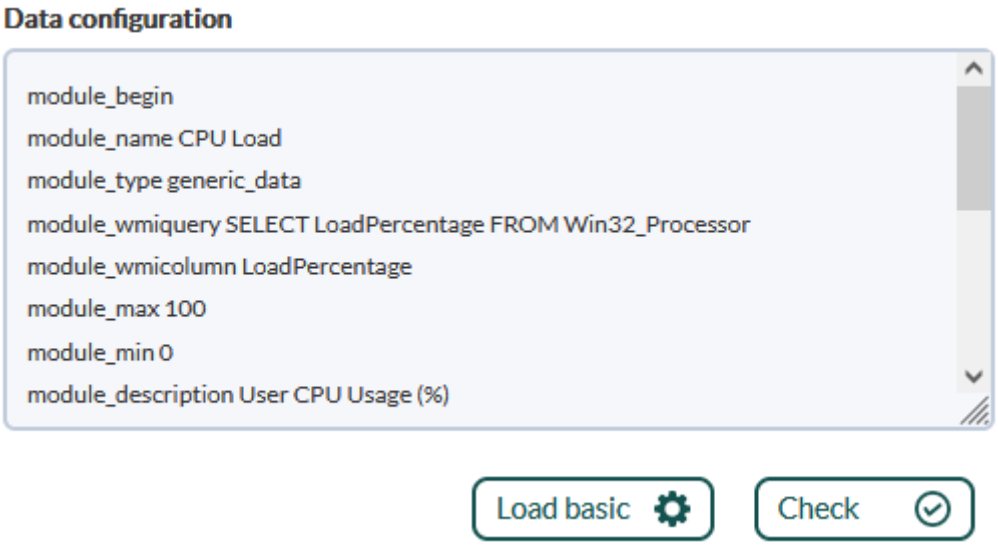
Intervalos en los módulos locales

Los módulos locales (o de agente software) tienen todos como “base” el intervalo de su agente. Sin embargo, pueden tomar valores múltiplos de esa base si modifica el parámetro `module_interval` con un multiplicar entero mayor que cero.

Interfaz de creación de módulos

La configuración remota del Agente Software respectivo
debe estar habilitada.

La creación de módulos locales en la consola se realiza mediante un formulario donde, además de la configuración común de todo módulo (umbrales, tipo, grupo, etcétera) dispone de una caja de texto donde especificar los datos de configuración a establecer en el fichero de configuración del Agente Software.



- Al hacer clic en el botón Load basic (template), se borrará el contenido de *Data configuration* con una plantilla básica que deberemos modificar de acuerdo a la necesidad de monitorización.
- Una vez modificado, al hacer clic en Check (syntax) verificará que la sintaxis de plantilla siga siendo correcta, sin embargo, el resto de los comando no serán comprobados.

Cuando un módulo es cargado desde un componente local, puede tener macros. Si tiene macros, la caja de configuración estará oculta y aparecerá un campo por cada macro, ver más información en Plantillas y componentes

Monitorización condicionada

Postcondiciones

El Agente Software soporta la ejecución de comandos y *scripts* en modo de postcondiciones. Esto quiere decir que se pueden realizar acciones dependiendo del valor obtenido en la ejecución del módulo. El parámetro module_condition se utiliza para ello, por ejemplo: `module_condition < 20 add_processes.sh`.

Precondiciones

El parámetro `module_precondition` permite evaluar una condición antes de la ejecución del módulo y con el resultado decidir si el módulo se debe ejecutar o no, por ejemplo: `module_precondition> 10 number_active_processes.sh` .

Monitorización intensiva

Existen ciertos módulos que tienen una importancia especial, tales como procesos o servicios críticos en ejecución. Para poder tener una monitorización más controlada de estos casos existe la monitorización intensiva.

Consiste en avisar en un intervalo más corto de que ha aparecido un problema serio sin necesidad de reducir el intervalo general del agente.

Configuración en Agente Software:

- interval: Obligatorio, tiempo de muestreo del agente en segundos, es el intervalo general para todos los módulos locales.
- intensive_interval: Tiempo en que avisará si existe algún problema, y siempre se ejecutará en este período y si coincide con la condición se notificará en este período de tiempo (de lo contrario los datos se enviarán el el interval).

Configuración en módulo:

- module_intensive_condition = <valor>: si el módulo obtiene como resultado el <valor> indicado en este parámetro, notificará en el intervalo intensivo antes definido. Otros operadores que se pueden utilizar son: <, >, `!=`, un rango de valores `(m,n)` y =~.

Ejemplo

El servicio sshd es muy importante pues es utilizado para conectar por *shell* de manera remota, necesitamos monitorizar su funcionamiento:

```
intensive_interval 10
interval 300
```

```
module_begin
module_name SSH Daemon
module_type generic_data
module exec ps aux | grep sshd | grep -v grep | wc -l
module_intensive_condition = 0
module_end
```

Si el servicio está ausente, se notificará en los próximos 10 segundos, si está funcionando notificará cada 5 minutos (intervalo normal, 300 segundos).

Monitorización programada

El Agente Software soporta la definición de módulos programados que se ejecutan en los instantes definidos. La sintaxis usada es la misma que la del fichero crontab.

Chequeos remotos con el agente software

Un Agente Software es capaz de realizar chequeos remotos, sustituyendo el servidor principal PFMS e incluso distribuirlos en *agentes broker*

Chequeos ICMP

Los chequeos ICMP o pingson muy útiles para saber si una máquina está conectada o no a una red.

Unix

```
module_exec ping -c 1 dir_IP> /dev/null 2>&1; if [ $? -eq 0 ]; then echo 1; else echo 0; fi
```

MS Windows®.

```
module_ping dir_IP
```

Nota: module_advanced_optionspermite opciones avanzadas para `ping.exe`.

Chequeos TCP

Los chequeos TCP son útiles para verificar que los puertos de una máquina permanecen abiertos y permiten conocer si una aplicación conecta o no a la red.

Unix

Con el comando nmap y sus parámetros de configuración en la línea de comando, a una dirección IP chequeamos si el puerto 80 está abierto (tiempo de espera de respuesta de 5 segundos):

```
module_begin
module_name PortOpen
module_type generic_proc
module_exec nmap 192.168.100.54 -p 80 | grep open > /dev/null 2>&1; echo $?; if
[ $? == 0 ]; then echo 1; else echo 0; fi
module_timeout 5
module_end
```

MS Windows®.

Los parámetros se deben especificar en:

- module_tcpcheck: dirección IP del dispositivo.
- module_port: número de puerto.
- module_timeout: tiempo de espera para la respuesta.

Ejemplo:

```
module_begin
module_name TcpCheck
module_type generic_proc
module_tcpcheck 192.168.100.54
module_port 80
module_timeout 5
module_end
```

Chequeos SNMP

Los chequeos SNMP son comunes en la monitorización de dispositivos de red para comprobar el estado de interfaces, bytes de entrada/salida, etc.

Ejemplo en Unix

```
module_exec snmpget dir_IP -v 1 -c public .1.3.6.1.2.1.2.2.1.1.148 | awk '{print
$4}'
```

Ejemplo en MS Windows®

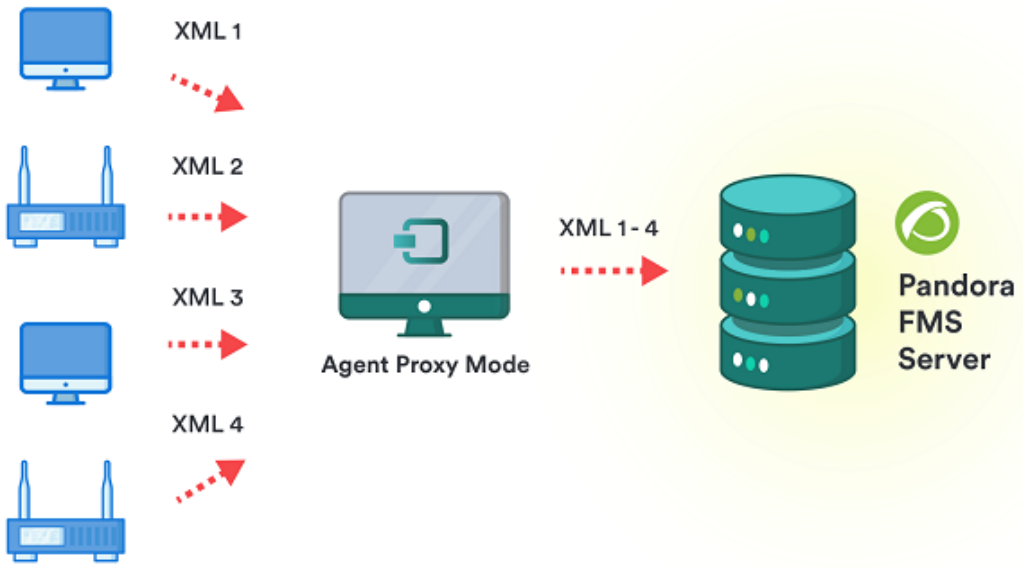
```
module_snmpget
module_snmpversion 1
module_snmp_community public
module_snmp_agent 192.168.100.54
```

```
module_snmp_oid .1.3.6.1.2.1.2.2.1.1.148
module_end
```

Modo Proxy

Para usar el modo *proxy* del agente de Pandora FMS en Linux/Unix® no puede ser ejecutado por el usuario root, por ello es necesario una instalación especial del agente de Pandora FMS. Para ello consulte la Instalación personalizada del Agente.

Este modo permite redirigir los ficheros de datos generados por otros Agentes Software al servidor de Pandora FMS. El agente software que actúa en Modo Proxy también puede realizar tareas de monitorización.

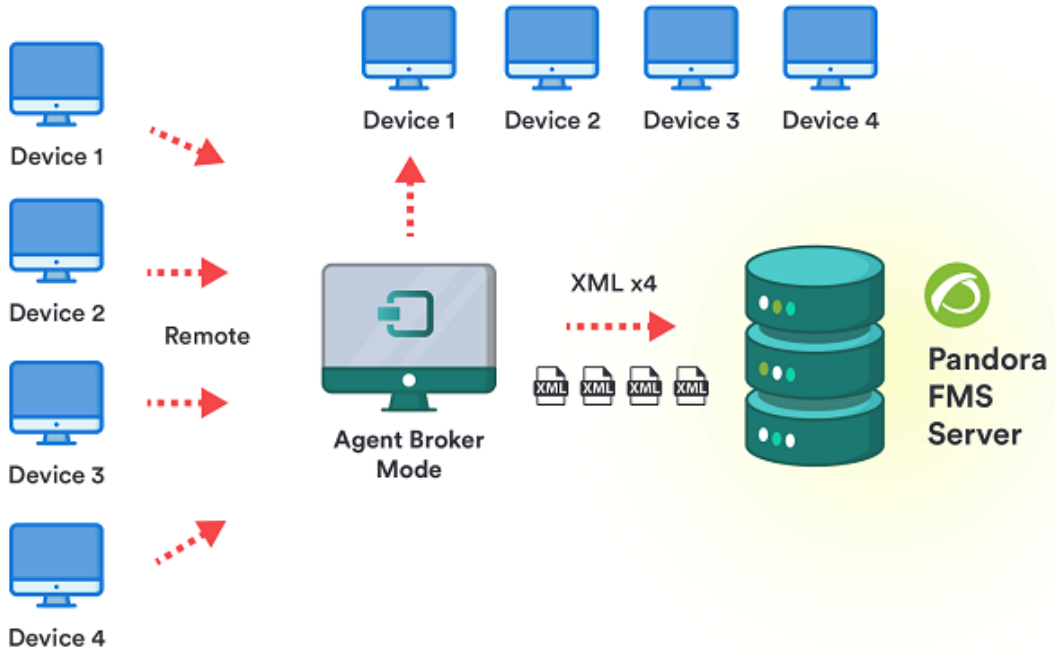


Configuración de los parámetros:

- server_ip: Dirección IP Pandora FMS server.
- proxy_mode: activado (1) o desactivado (0).
- proxy_max_connection: número de conexiones simultáneas del *proxy*, por defecto 10.
- proxy_timeout: tiempo de espera de respuesta para el proxy, por defecto 1 segundo.
- proxy_address: dirección en la que escucha el *proxy*.
- proxy_port: puerto en el que escucha el *proxy*.

Modo Broker

EL Modo Broker de los Agentes Software permite a un solo agente realizar chequeos y gestionar la configuración como si se tratara de varios agentes distintos.



Cuando se activa el Modo Broker en un Agente Software, se crea un nuevo fichero de configuración. A partir de ese momento, el Agente Software original y el nuevo broker se gestionarán de forma separada con sus ficheros de configuración independientes, como si fuesen dos Agentes Software totalmente separados en la misma máquina.

Para crear un Broker se añade una o más líneas con el parámetro broker_agent <nombre_broker> (una línea para cada Broker).

En la Consola web de Pandora FMS los Broker se ven y gestionan como agentes independientes.

- Los módulos que guardan datos en memoria entre ejecuciones (`module_logevent` y `module_regexp` en MS Windows®) no funcionan cuando hay agentes broker configurados.
- Las instancias del modo Broker no pueden utilizar colecciones.

Inventario con agente software

Para más información visite la sección Inventario local con agentes software.

Recolección de logs con agente software

Para más información visite el tema Recolección y monitorización de logs.

Acciones remotas por UDP

Un Agente Software es capaz de recibir peticiones remotas y ejecutar órdenes.

Tenga presente que UDP es por naturaleza inseguro (pero eficiente para enviar mensajes sin comprometer una respuesta cierta).

Para permitir que el servidor PFMS envie órdenes a los Agentes Software a su cargo, se debe configurar:

- udp_server: cero por defecto, valor en uno (1) para activar esta funcionalidad.
- udp_server_port: puerto de escucha en Agente Software.
- udp_server_auth_address: dirección IP del servidor Pandora FMS

Reinicie el Agente Software para que los cambios se apliquen.

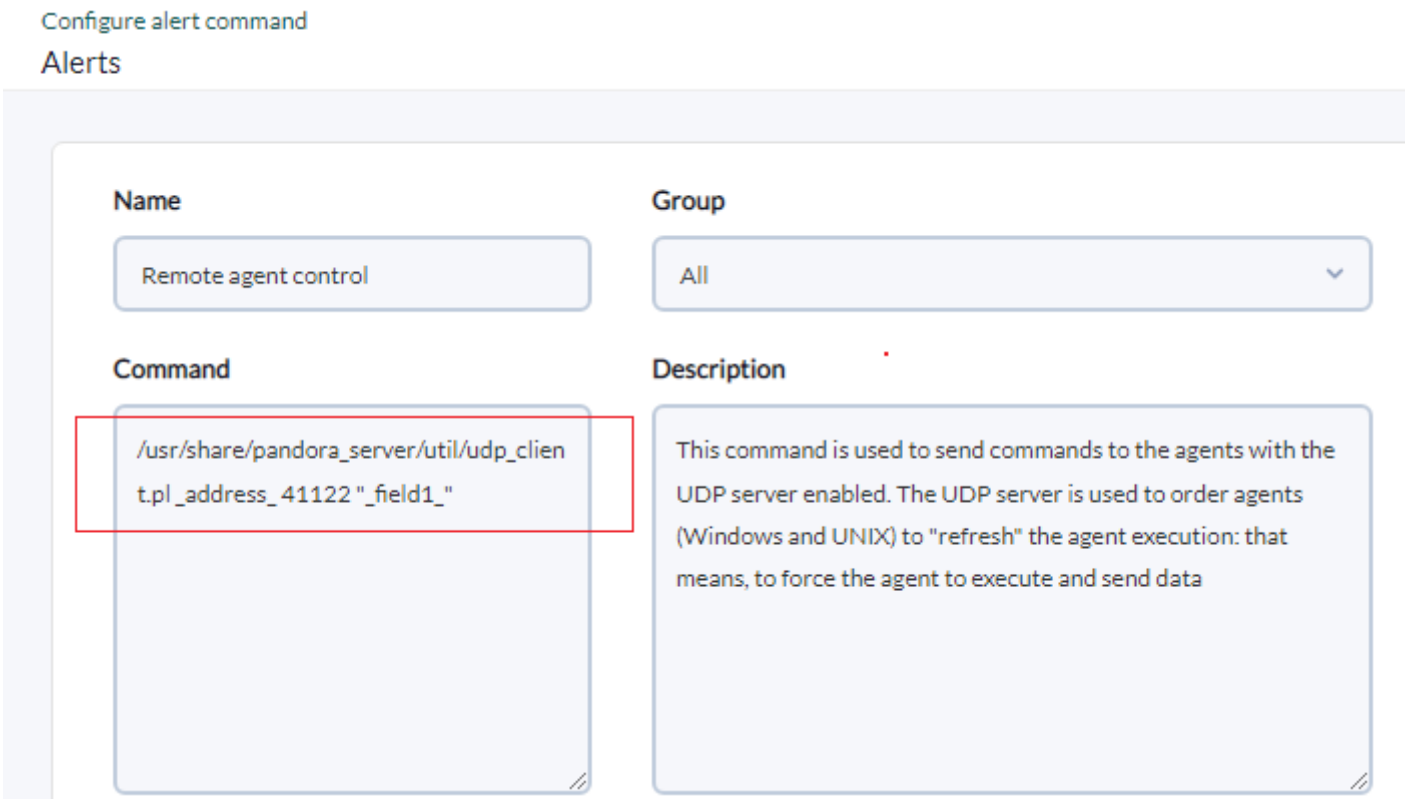
Aunque puede establecerse a *0.0.0.0* para que acepte desde todos los orígenes, dicha práctica no es recomendada. Si tiene varios Servidores PFMS y/o utiliza IPv6 puede colocar diferentes direcciones IP separadas por comas. Por ejemplo si tiene en IPv6:`2001:0db8:0000:130F:0000:0000:087C:140B` y su abreviatura es `2001:0db8:0:130F::87C:140B` utilice ambas separadas por comas.

Cómo solicitar reinicio de servicio de Agentes Software

Se debe utilizar el *script* ubicado en:

```
/usr/share/pandora_server/util/udp_client.pl
```

Se puede ejecutar desde línea de comando o bien utilizarlo en una alerta, mediante el comando que viene preconfigurado en la consola Remote agent control.



Acciones remotas personalizadas

Además de la acción de reiniciar servicio de Agente Software, se pueden especificar acciones personalizadas.

```
process_<order_name>_start comando
```

También se pueden crear órdenes que llamen a *scripts* para realizar múltiples acciones remotas con solamente pulsar un botón.

Plugins en agentes software

A diferencia de los *plugins* de servidor, ejecutados por Pandora FMS server, los *plugins* de Agente Software reportan uno o varios módulos a la vez.

Ejecución en sistemas Windows

En MS Windows®, todos los *plugins* por registrados por defecto están programados en VBScript, para ejecutarlos se utiliza el intérprete `cscript.exe`.

Chequeos utilizando PowerShell

A partir de la versión 776 se cuenta con `module_exec_powershell` el cual permite introducir comandos más complejos en PowerShell con caracteres especiales e instrucciones complejas (una instrucción entrega resultados a la siguiente) que no son posibles mediante el módulo `module_exec`.

```
# Example of Powershell execution module
module_begin
module_name Powershell
module_type generic_data_string
module_exec_powershell < command_1 > | < command_2 > | … | < command_N >
module_end
```

Los comandos se introducen tal cual, sin necesidad de comillas para que sean procesados por el Agente Software PFMS (los comandos de PowerShell, en cambio, sí pueden necesitar de comillas.) Si el comando no es válido se agrega un error al registro del agente (fichero `pandora_agent.log`).

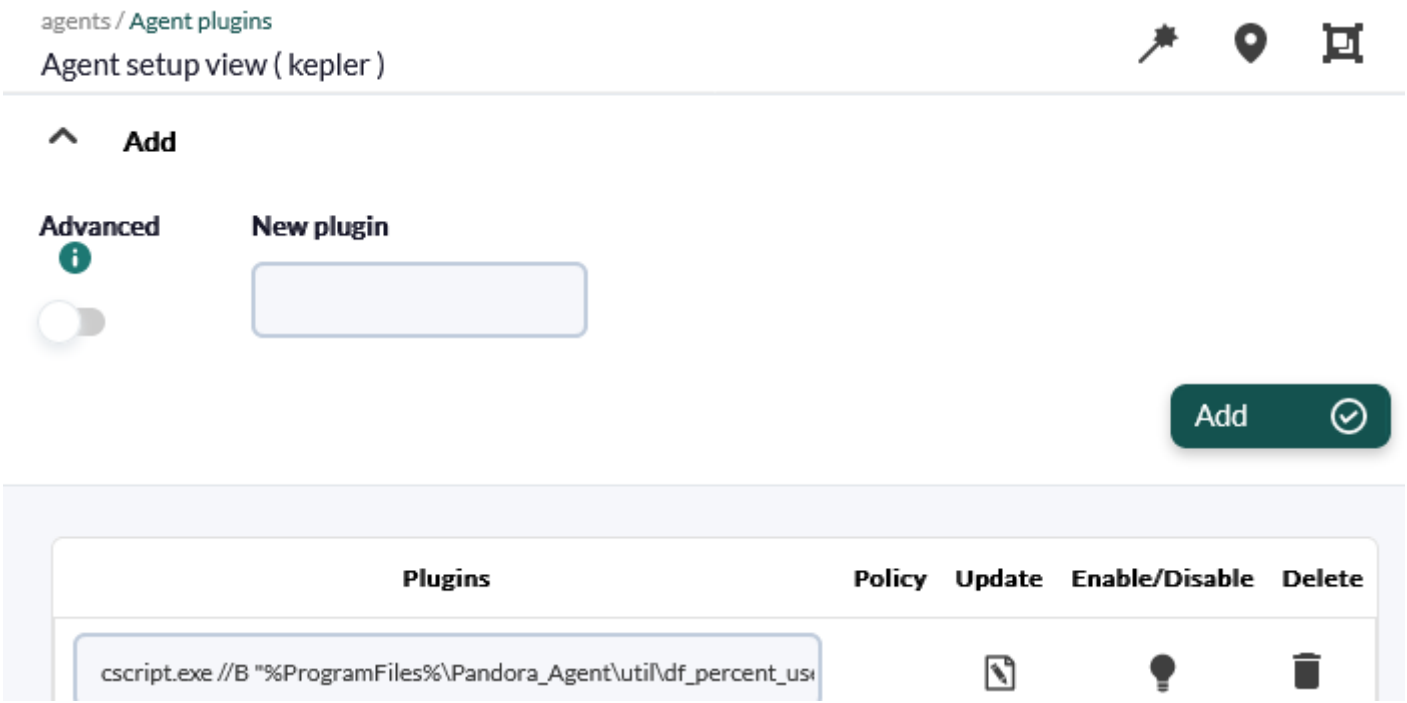
Ejecución en sistemas Unix

Los *plugins* de Unix estén ubicados por defecto en el directorio del agente:

```
/etc/pandora/plugins
```

Gestión de plugins de Agente Software desde la Consola

Al tener su configuración remota activada, un Agente Software en su vista de administración dispondrá de la pestaña del editor de plugins.



Gestión de plugins avanzados de Agente Software desde la Consola

Es posible agregar un *token* en la configuración de los *plugins* de agente que al ser habilitado permite la opción de 'encapsular' las definiciones de *plugins* dentro de las etiquetas `module_begin` y `module_end`.

Este *token* habilitado permite insertar bloques de configuración como `module_interval` o `module_crontab`, entre otros.

Cómo crear plugins personalizados para Agente Software

Los *plugin* pueden ser creados en cualquier lenguaje de programación. Solo se debe tener en cuenta las normas generales y las normas específicas para su desarrollo.

Asegúrese de terminar la salida del nuevo *plugin* (si es un *script*) con un `errorlevel 0` o el agente interpretará que el *plugin* ha tenido un error y no ha podido ejecutar el trabajo.

Utilizando plugins de Nagios desde Agente Software

Nagios tiene un gran número de *plugins* que puede utilizar con Pandora FMS. Un modo de hacerlo es utilizar los *plugins* remotos con el Plugin Server, usando la compatibilidad de Nagios.

Monitorización con KeepAlive

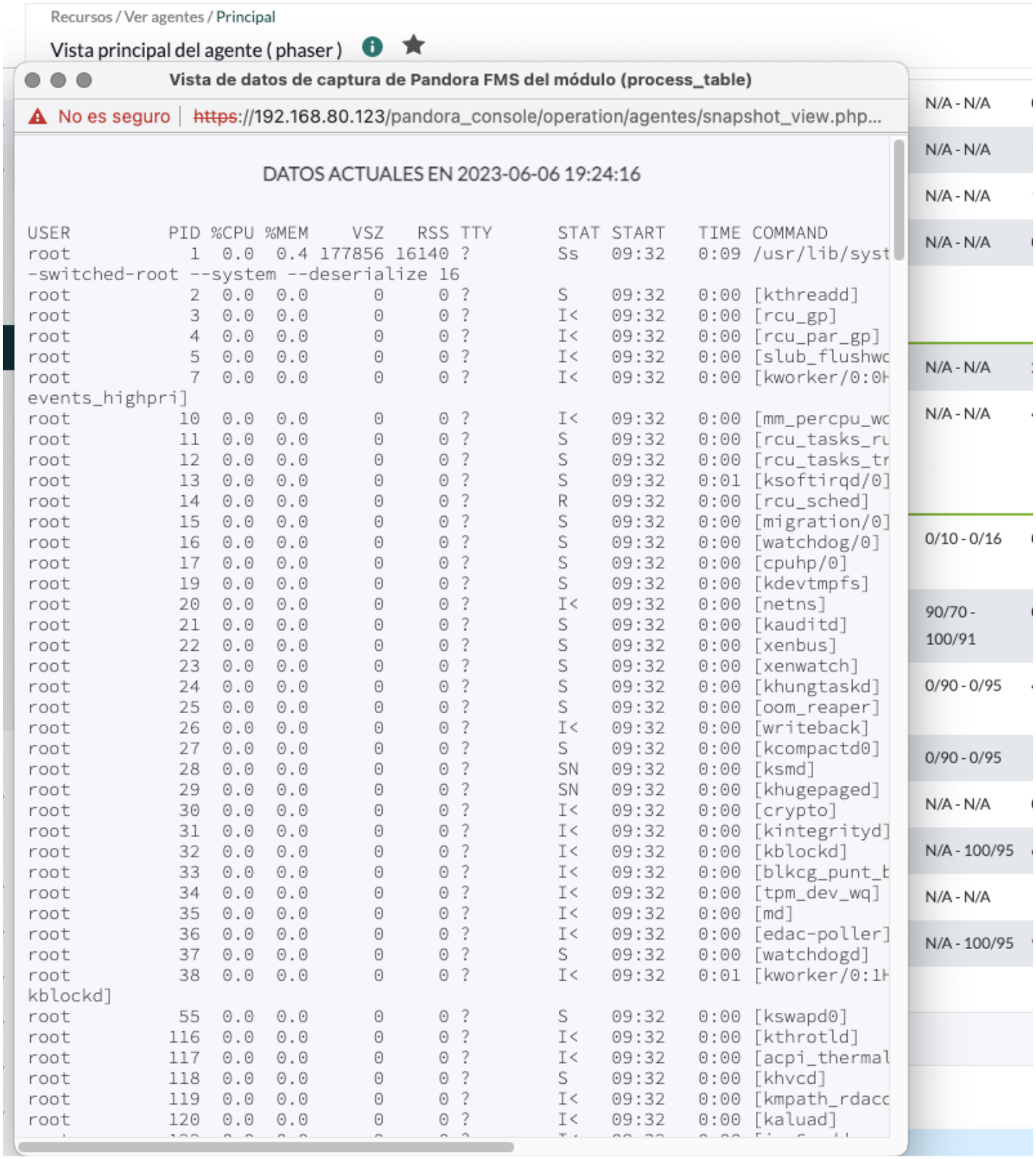
El módulo KeepAlive solamente se puede crear desde Consola, aunque no se tenga configuración remota habilitada y no deja ninguna traza en el fichero `pandora_agent.conf`.

Un módulo singular en Pandora FMS es el tipo llamado `keep_alive`, utilizado para alertar si un Agente Software ha dejado de enviar información.

Se debe ir a la pestaña de modulos (Management → Manage agents → clic en nombre de agente → Modules).

Se debe pulsar Create module y seleccionar Create a new data server module → Create → introducir nombre del nuevo módulo → Create.

Monitorización de capturas de comandos (Command snapshots)



Comandos que presenten salidas extensas, como `top` o `netstat -n` pueden ser capturados completamente por un módulo y reproducidos tal cual. El módulo debe configurarse como tipo texto, ejemplo:

```
module_begin
module_name process_table
module_type generic_data_string
module_exec ps aux
module_description Command snapshot of running processes
module_group System
module_end
```

- Para que esto funcione así, hay que configurar adecuadamente tanto la consola de Pandora (*setup*) como el agente que recoge esa información, asegurándose de que es *texto sin tratar*.
- En la Consola, se debe activar la opción Command line snapshot.

Monitorización y visualización de imágenes

Este método permite definir módulos de tipo cadena (`generic_data_string` o `async_string`) que contengan imágenes en formato texto con una codificación base64, pudiendo mostrar dicha imagen en lugar de un resultado concreto.

Por ejemplo:

```
#!/bin/bash
echo "<module>"
echo "<name>Actual leader</name>"
echo "<type>async_string</type>"
echo "<data><![CDATA[....]]></data>"
echo "</module>"
```

Se graba ese contenido a un archivo en el agente (o distribuya por colecciones) y ejecútelo así:

```
module_plugin <complete path to the file>
```

Monitorización específica para Windows

- Si el nombre del proceso contiene espacios en blanco no use “ ” .
- El nombre del proceso debe ser el mismo que muestra el Administrador de Tareas (`taskmngr`) de Windows, incluyendo la extensión `.exe`.
- Es importante respetar mayúsculas y las minúsculas.

Monitorización de procesos y watchdog de procesos

Monitorización de procesos

El parámetro module_proc comprueba si un determinado nombre de proceso está operando en esta máquina. Ejemplo:

```
module_begin
module_name CMDProcess
```

```
module_type generic_proc
module_proc cmd.exe
module_description Process Command line
module_end
```

Se debe añadir el parámetro module_async yes:

```
module_begin
module_name CMDProcess
module_type generic_proc
module_proc cmd.exe
module_async yes
module_description Process Command line
module_end
```

Watchdog sobre procesos

La funcionalidad de Watchdog para MS Windows® permite iniciar de nuevo un proceso interrumpido, ejemplo:

```
module_begin
module_name Notepad
module_type generic_data
module_proc notepad.exe
module_description Notepad
module_async yes
module_watchdog yes
module_user_session yes
module_start_command "%SystemRoot%\notepad.exe"
module_startdelay 3000
module_retrydelay 2000
module_retries 5
module_end
```

Monitorización de servicios y watchdog de servicios

Monitorización de servicios

El parámetro module_service comprueba si un determinado servicio se está ejecutando en la máquina. La definición de un módulo usando este parámetro sería:

```
module_begin
module_name Service_Dhcp
module_type generic_proc
module_service Dhcp
module_description Service DHCP Client
```

```
module_end
```

Para que avise inmediatamente cuando un proceso deja de funcionar se debe añadir el parámetro module_async yes (ver normas comunes al principio de sección Windows):

```
module_begin
module_name Service_Dhcp
module_type generic_proc
module_service Dhcp
module_description Service DHCP Client
module_async yes
module_end
```

Watchdog de servicios

Funciona de manera similar al Watchdog de procesos. Ejemplo:

```
module_begin
module_name ServiceSched
module_type generic_proc
module_service Schedule
module_description Service Task scheduler
module_async yes
module_watchdog yes
module_end
```

La definición del watchdog para servicios no requiere ningún parámetro adicional como el de procesos, porque esa información ya está dentro de la definición del servicio.

Monitorización de recursos básicos

Al instalar el Agente Software PFMS para MS Windows® se incluyen los módulos básicos necesarios, algunos vienen activos y otros se deben activar por Remote Configuration (o editando de manera local el fichero `.conf` del agente).

Volver al índice de documentación de Pandora FMS