



Monitorización de muchas máquinas de forma rápida



m:
<https://pandorafms.com/manual/!current/>
manent link:
https://pandorafms.com/manual/!current/es/documentation/pandorafms/technical_annexes/33_pfms_fast_deployment
4/06/10 14:36





Monitorización de muchas máquinas de forma rápida

Introducción

Esta guía pretende mostrar al usuario como administrar de forma rápida y eficiente un número elevado de máquinas (5,10,50,500...) utilizando las diferentes características de Pandora FMS diseñadas para este propósito. Dividiremos el documento en cuatro partes:

- Monitorización de dispositivos de red, usando Recon Server y plantillas.
- Monitorización de dispositivos de red SNMP, usando Recon Script SNMP.
- Monitorización de agentes, usando políticas.
- Monitorización remota con scripts personalizados, usando un generador de agentes vía XML.

Monitorización de dispositivos de red, usando Recon Server y plantillas

Situación

Tenemos que monitorizar 200 servidores, 20 switches y 10 routers, y no podemos ir uno por uno configurándolos. La monitorización “general” es muy sencilla, pero no tenemos mucho tiempo ni posibilidad de instalar agentes en las máquinas.

Solución

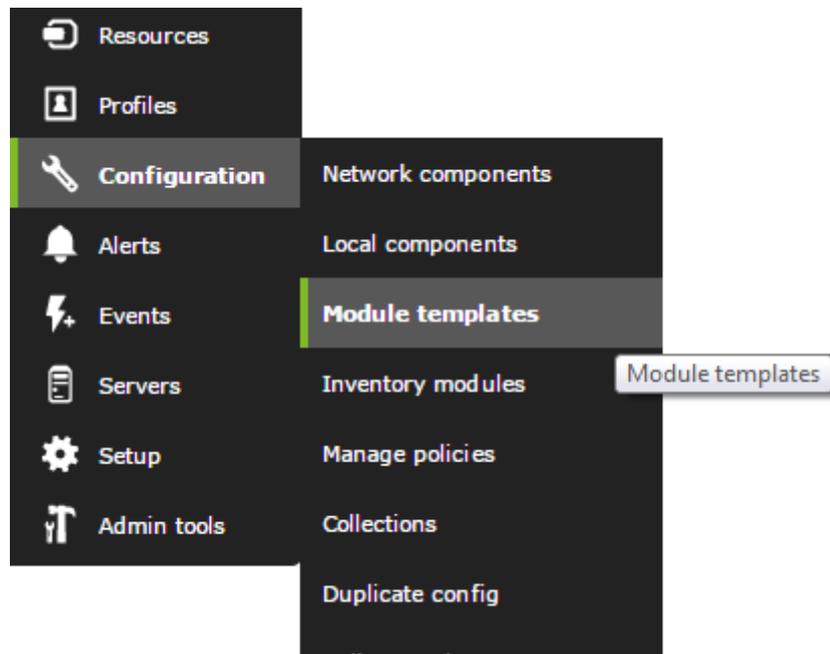
Pandora FMS detectará los sistemas y les aplicará diferentes plantillas en función de si es un switch, un router o un servidor. Las plantillas llevarán chequeos remotos que se puedan aplicar nada más detectar el tipo de máquina.

¿Cuánto tardaré?

Una red de clase C (255 hosts) se escanea en menos de un minuto con la versión 6.0. Aplicar a las máquinas detectadas un patrón de monitorización es casi inmediato, con lo que puede tener esas 230 máquinas completamente configuradas en menos de diez minutos.

Paso 1. Definir los perfiles de monitorización

Primero vamos a definir una plantilla de monitorización que en Pandora FMS se llama “Module template”. Para ello vamos al siguiente menú:



Aquí veremos unos perfiles ya definidos, que contienen algunos chequeos genéricos. Vamos a editar uno de ellos (Linux Server) que hace referencia a un perfil útil para monitorizar servidores Linux genéricos de forma remota.

Module management » Module management

Name	Description	Action
Basic DMZ Server monitoring	This group of network checks, ch[...]es located on DMZ servers...	
Basic Monitoring	Only checks for availability and latency of targeted hosts.	
Linux Server with SNMP	Group of "basic" modules for SNM[...]s and a full range of System	

Create > Delete

Module management » Module management

Name:

Description:

Create

Como se puede ver en la captura superior, este perfil tiene algunos chequeos TCP básicos, como por ejemplo "Check SSH Server", un chequeo ICMP básico: "Host Alive" y diversos módulos SNMP que hacen uso de la MIB de Linux, que son el resto de chequeos.

Estos chequeos "de plantilla" están definidos en la biblioteca de módulos básica de Pandora FMS, y contienen definiciones genéricas de módulos.

El valor IP no existe en este módulo, porque se autoasignará de la IP del agente. El resto de campos son "por defecto", p.e: umbrales, comunidad SNMP, y se aplicarán a todos los agentes

que tengan una plantilla con este módulo. Si queremos personalizarla (p.e: cambiar la comunidad) habrá que cambiarla en los agentes uno por uno o de forma general con la herramienta de cambios masivos.

Ahora que ya sabemos lo que es una plantilla de monitorización y un módulo genérico para plantilla, podemos ver algunas de las otras plantillas, concretamente la de monitorización genérica WMI y la de monitorización básica.

La primera contiene tres módulos WMI para Windows. Estos módulos habrá que personalizarlos, editando el componente original o los módulos generados, ya que requieren usuario y password con permisos para hacer consultas remotas WMI.

La segunda solo contiene un chequeo básico de conectividad ICMP, y podemos agregar otros chequeos básicos tal y como vemos en la siguiente captura:

F.	P.	Type	Module name	Description	Status	Warn	Data	Graph	Last contact
			Connections opened	Network connections used in this machine	 	0/100 - 0/150	439 conns		7 minutes 25 seconds
			CPU Usage	% of CPU usage in this machine	 	0/60 - 0/90	10 %		7 minutes 25 seconds
			Disk_Free	Disk space available in MB.	 	20/10 - 10/0	35.0 MB		7 minutes 25 seconds
			Dropped Bits of nothing	Simulation of big number with absolute nonsense, real like li...	 	N/A - N/A	317,615,070 gamusins		7 minutes 26 seconds
			Memory_free		 	N/A - 50/0	7,869.2 MB		7 minutes 26 seconds
			Network Traffic (Incoming)	Network throughput for incoming data	 	N/A - 0/900k	764,725 kbit/sec		7 minutes 26 seconds
			Network Traffic (Outgoing)	Network throughput for Outgoing data	 	N/A - 0/900k	385,559 kbit/sec		7 minutes 26 seconds
			Server Status A	Status of my super-important daemon / service / process	 	N/A - N/A	11		7 minutes 26 seconds
			Server Status B	Status of my super-important daemon / service / process	 	N/A - N/A	78		7 minutes 26 seconds
			Server Status C	Status of my super-important daemon / service / process	 	N/A - N/A	39		7 minutes 26 seconds
			System Log File	Messages from the system in logfile format	 	N/A - N/A	HWTbUZwsgBDL		7 minutes 26 seconds

Paso 2. Utilizar una tarea de red con Recon Server

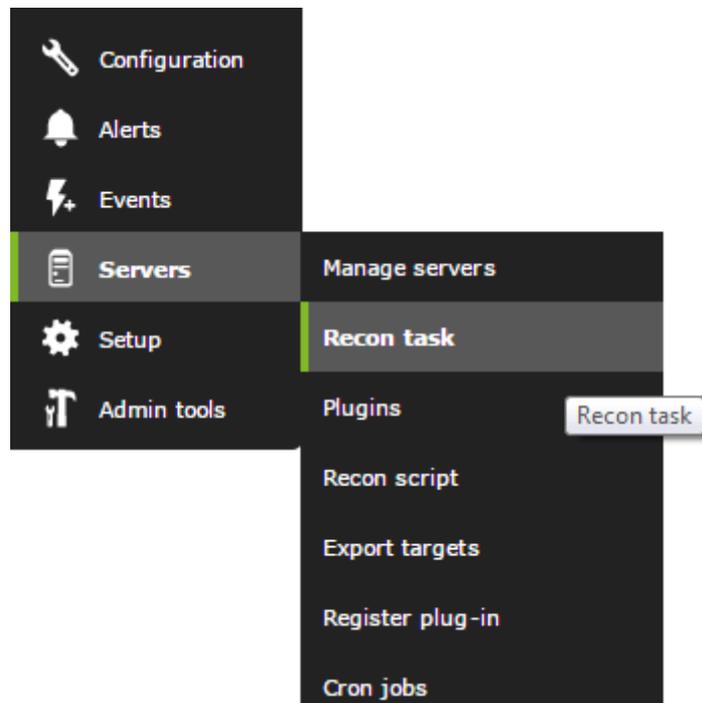
Ahora que tenemos tres perfiles básicos de monitorización: Linux, Windows y red.

Supongamos que tenemos que monitorizar todos los equipos en un conjunto de redes, por ejemplo:

- 192.168.50.0/24 para servidores.
- 192.168.50.0/24,192.168.1.0/24 para comunicaciones.

Y queremos que identifique todas las máquinas de esas redes y en función de su SO le aplique una plantilla u otra. Otra forma de hacerlo, ya que los switches pueden ser de varias marcas y modelos, es "identificarlos" por medio de un patrón basado en tener o no un puerto abierto. P.e: que aquellas máquinas con el puerto 23 (telnet) abierto las identifique como máquinas genéricas (switches, routers).

Vamos a la sección de Recon Servers para crear uno nuevo:



Vamos a crear uno para buscar y dar de alta servidores Windows, aplicándole el patrón de monitorización de máquinas Windows:

Manage recontask ?

Task name	Windows Server
Recon server *	catan
Mode	Network sweep
Network	192.168.50.0/24
Interval *	Manual
Module template	Basic DMZ Serv... monitoring
OS	Windows
Ports	
Group	Applications
Incident	Yes *
SNMP Default community	public
Comments	
OS detection	<input checked="" type="checkbox"/>
Name resolution	<input checked="" type="checkbox"/>
Parent detection	<input checked="" type="checkbox"/>
Parent recursion	5 *

Add

Aquí se puede ver como en el campo "OS" (tipo de sistema operativo), hemos elegido Windows por lo que solo aplicará este perfil de monitorización a aquellas máquinas que sean de tipo Windows, y en caso contrario serán ignoradas. Dado que la forma de detectar automáticamente el tipo de SO no es 100% fiable (depende de los servicios de la propia máquina), se podría escoger otro método, como especificar un puerto concreto.

De esa forma, todas las máquinas con ese puerto abierto, entrarían en la aplicación de la plantilla. Ese ejemplo lo vemos aquí, donde hemos creado otra tarea pero usando un filtrado por puerto en vez de por SO para aplicarle la plantilla de monitorización de dispositivos de red genéricos:

Manage recontask ?

Task name	<input type="text" value="Windows Server"/>
Recon server ★	<input type="text" value="catan"/>
Mode	<input type="text" value="Network sweep"/>
Network	<input type="text" value="192.168.50.0/24"/>
Interval ★	<input type="text" value="Manual"/>
Module template	<input type="text" value="Basic DMZ Serv... monitoring"/>
OS	<input type="text" value="Any"/>
Ports	<input type="text" value=""/>
Group	<input type="text" value="Applications"/>
Incident	<input type="text" value="Yes"/>
SNMP Default community	<input type="text" value="public"/>
Comments	<input type="text" value=""/>
OS detection	<input checked="" type="checkbox"/>
Name resolution	<input checked="" type="checkbox"/>
Parent detection	<input checked="" type="checkbox"/>
Parent recursion	<input type="text" value="5"/>

Add ➤

Es importante fijarse también en que para especificar dos redes, hay que separarlas por comas: 192.168.50.0/24,192.168.1.0/24

Por último configuraría el de Linux de forma similar, y al terminar de definir los tres grupos quedaría de la siguiente manera:

Manage recontask 👁

✔

SUCCESS

Successfully created recon task

✕

Name	Network	Mode	Group	Incident	OS	Interval	Ports	Action
Prueba	216.58.211.0/22	<input checked="" type="checkbox"/>		Yes	Any	Manual		
Windows Server	192.168.50.0/24	<input checked="" type="checkbox"/>		Yes		1 days		
Any Server	192.168.50.0/24	<input checked="" type="checkbox"/>		Yes	Any	Manual		

Create ➤

Una vez definidas las tareas de reconocimiento, estas pueden empezar solas, pero vamos a ver su estado y a forzarlas si fuera necesario. Para eso, haremos click en el icono del ojo, para ir a la vista de operación del servidor Recon.

Recon View								
Force	Task name	Interval	Network	Status	Template	Progress	Updated at	Edit
<input type="radio"/>	Prueba	Now	216.58.211.0/22	Done		-	1 days	
<input type="radio"/>	Windows Server	1 days	192.168.50.0/24	Done		-	8 minutes 45 seconds	
<input type="radio"/>	Any Server	Now	192.168.50.0/24	Pending		<div style="width: 20%;"><div style="background-color: #007bff; height: 10px;"></div></div> 20%	2 seconds	

Por defecto el servidor de reconocimiento (recon_server) tiene un hilo de ejecución, por lo que podrá ejecutar solo una tarea a la vez, el resto esperará a que termine la tarea de exploración activa; no obstante esto es modificable en el fichero de configuración del servidor (pandora_server.conf). Podemos forzar las tareas de exploración pulsando el icono verde redondo a la izquierda de la tarea.

Esto hará que el servidor recon busque máquinas nuevas que no existan en la monitorización activa. Si las encuentra, las dará de alta automáticamente (intentando resolver el nombre, si hemos activado esa opción) y asignándole todos los módulos que estaban contenidos en el perfil.

Debemos ser conscientes de que muchos de los módulos asignados en un perfil pueden no tener sentido o no estar correctamente configurados para un agente en concreto. En este agente, hemos detectado un sistema Linux correctamente, pero ese servidor no tiene SNMP, por lo que todos los modulos SNMP no están reportando. Dado que ni siquiera la primera vez pudieron obtener datos, están en un modo conocido como “estado Non-init” (no inicializado). La próxima vez que pase el script de mantenimiento de la BBDD, se eliminarán automáticamente:

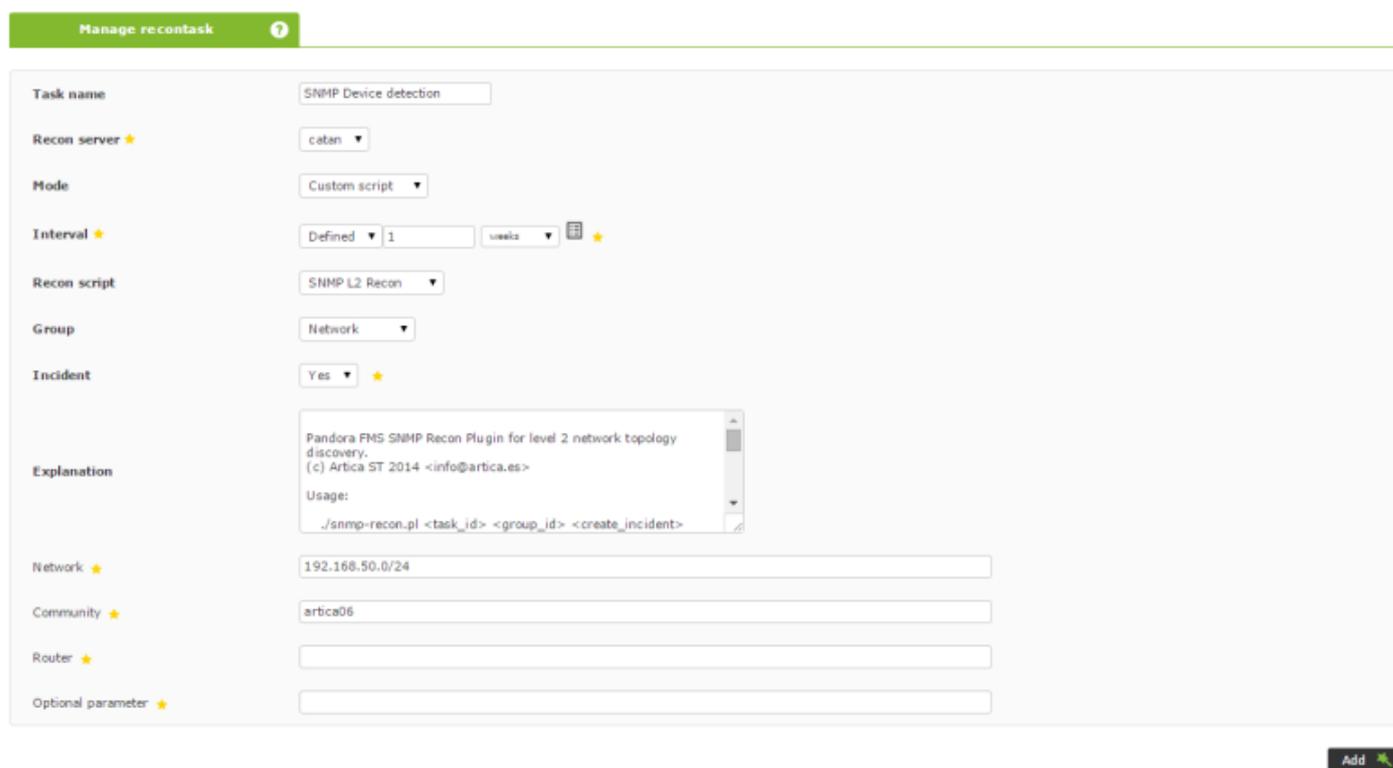
Name	P.	S.	Type	Interval	Description	Warn	Action
General							
Sysname			SNMP TEXT	900	Get name of[...]tandard MIB	N/A - N/A	<input type="checkbox"/>
Networking							
Check SSH Server			TCP PROC	300	Checks port 22 is opened	N/A - N/A	<input type="checkbox"/>
Host Alive			ICMP PROC	120	Check if ho[...]ping check.	N/A - N/A	<input type="checkbox"/>
NIC #1 in0ctects			SNMP INC	180	Input troug[...]interface #1	N/A - N/A	<input type="checkbox"/>
NIC #1 out0ctects			SNMP INC	180	Output thro[...]interface #1	N/A - N/A	<input type="checkbox"/>
NIC #1 status			SNMP PROC	180	Status of NIC#1	N/A - N/A	<input type="checkbox"/>
System							
OS CPU Load (1 min)			SNMP DATA	180	CPU Load in[...] (UNIX MIB)	N/A - N/A	<input type="checkbox"/>
OS CPU Load (5 min)			SNMP DATA	180	CPU load on[...] some UNIX)	N/A - N/A	<input type="checkbox"/>
OS IO Signals sent			SNMP INC	180	IO Signals sent by Kernel	N/A - N/A	<input type="checkbox"/>
OS Raw Interrupts			SNMP INC	180	Get system [...]pts from SO	N/A - N/A	<input type="checkbox"/>
OS Total process			SNMP DATA	180	Total proce[...] (UNIX MIB)	N/A - N/A	<input type="checkbox"/>
OS Users			SNMP DATA	180	Active user[...] (UNIX MIB)	N/A - N/A	<input type="checkbox"/>
System Description			SNMP TEXT	9000	Get system [...] (all mibs).	N/A - N/A	<input type="checkbox"/>
System Uptime			SNMP DATA	180	Sistem upti[...]n timeticks	N/A - N/A	<input type="checkbox"/>

Monitorización de dispositivos de red SNMP, usando Recon Script SNMP

En este escenario, nos planteamos la necesidad de monitorizar de forma “automática” a fondo un dispositivo SNMP con muchas interfaces, necesitando obtener el estado de cada interfaz, el tráfico en cada boca, la tasa de errores, etc.

Para ello, usaremos un sistema conocido como Recon Script. Es un sistema modular que permite ejecutar acciones complejas en un script. Pandora FMS dispone de un script ya creado para detectar este tipo de dispositivos SNMP.

Para ello, creamos una tarea de red, con la siguiente forma:



The screenshot shows the 'Manage reontask' configuration page in Pandora FMS. The task is named 'SNMP Device detection'. The configuration includes:

- Task name:** SNMP Device detection
- Recon server:** catan
- Mode:** Custom script
- Interval:** Defined, 1 weeks
- Recon script:** SNMP L2 Recon
- Group:** Network
- Incident:** Yes
- Explanation:** Pandora FMS SNMP Recon Plugin for level 2 network topology discovery. (c) Artica ST 2014 <info@artica.es> Usage: ./snmp-recon.pl <task_id> <group_id> <create_incident>
- Network:** 192.168.50.0/24
- Community:** artica06
- Router:** (empty field)
- Optional parameter:** (empty field)

An 'Add' button is visible at the bottom right of the configuration area.

En el “primer campo”, ponemos la red o redes de destino. En el “segundo campo”, ponemos la comunidad SNMP que vamos a emplear a la hora de explorar estos dispositivos. En el “tercer campo”, ponemos algunos parámetros opciones. En este caso -n es para que dé de alta también las interfaces caídas, ya que por defecto solo da de alta las interfaces activas.

Este script dará de alta las interfaces que antes no estaban y ahora están activas en cada máquina, en cada ejecución. De forma que si se levantan nuevas interfaces serán detectadas y añadidas. Las tareas de red pueden ser programadas para que se ejecuten de forma periódica, por ejemplo, una vez al día.

Este es el aspecto que tiene la tarea de tipo Task Recon Script una vez creada:

Name	Network	Mode	Group	Incident	OS	Interval	Ports	Action
SNMP Device detection	-	SNMP L2 Recon	-	Yes	-	7 days	-	  

Y este es el aspecto que tiene la tarea de tipo Task Recon Script en ejecución:

Force	Task name	Interval	Network	Status	Template	Progress	Updated at	Edit
<input checked="" type="radio"/>	SNMP Device detection	7 days	-	Pending	SNMP L2 Recon	<div style="width: 50%;"><div style="background-color: #007bff; height: 10px;"></div></div> 5%	1 minutes 27 seconds	

Monitorización de agentes mediante políticas

Para gestionar masivamente la monitorización de equipos con agente de software instalado nos valdremos de las políticas.

En primer lugar debemos tener los agentes de software ya instalados y con el parámetro `remote_config` habilitado, ya que de lo contrario no podremos crear módulos de ejecución:

```
remote_config 1
```

A continuación navegaremos hasta la sección de *Gestionar políticas*, y procederemos a crear una nueva política, completando algunos de los parámetros informativos como nombre, grupo y descripción:

ADD POLICY

Name

Group 

Description

Desde aquí podemos navegar a la sección de creación de módulos dentro de la política, y crear un nuevo módulo local (*dataserver module*):

FRESH NEW POLICY - MODULES

INFORMATION
There are no defined modules

Search Filter

Type

- Create a new data server module
- Create a new data server module
- Create a new network server module
- Create a new plug-in server module
- Create a new WMI server module
- Create a new webserver module

Create

Pandora FMS Library

Copy modules

Copy selected modules to policy :

Una vez creados tantos módulos como necesitemos, que pueden ser tanto de ejecución local (*dataserver module*) como de ejecución remota, podemos proceder a incluir en la política tantos agentes como queramos. Para ello navegaremos a la solapa correspondiente dentro de nuestra política, y moveremos agentes a la sección de “Agentes incluidos en la política”:

FRESH NEW POLICY - AGENTS

SUCCESS
Successfully added

Filter group Group recursion Filter agent

Agents

- 112_dev
- 192.168.50.2
- 192.168.50.3
- 192.168.50.4
- 192.168.50.5
- 192.168.50.6
- 192.168.50.10
- 192.168.50.12
- 192.168.50.14
- 192.168.50.18

Agents in Policy

- escoba
- esxi1
- ha-datacenter
- HADES

Agents

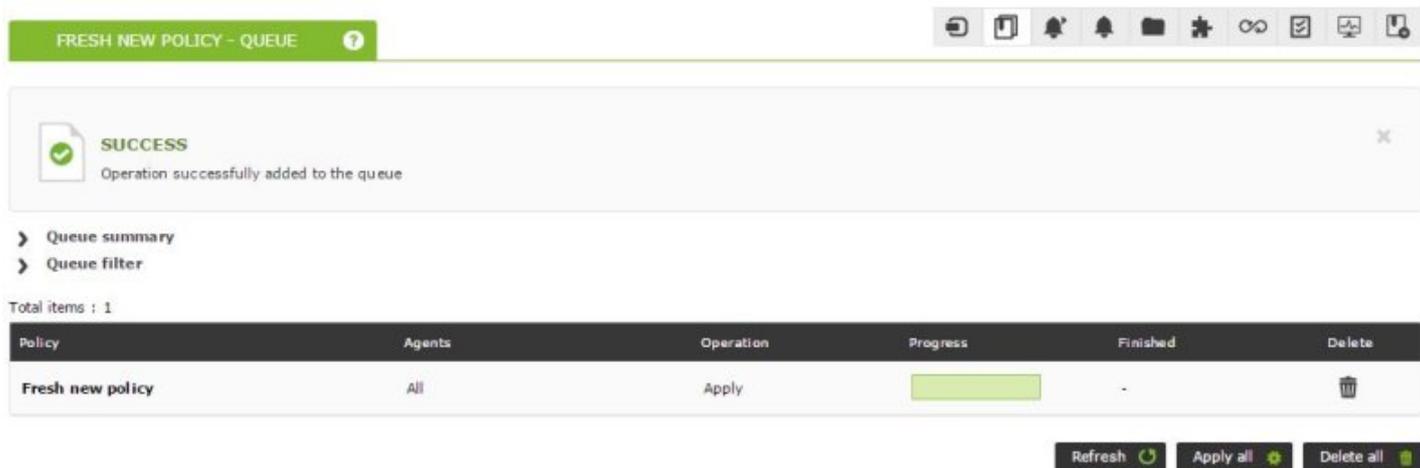
Group Group recursion Search

Applied Not applied All

Total items : 4

Name	R.	S.	U.	A.	Last application	D.
escoba	<input type="checkbox"/>	<input type="checkbox"/>	0	<input type="checkbox"/>		<input type="checkbox"/>
esxi1	<input type="checkbox"/>	<input type="checkbox"/>	0	<input type="checkbox"/>		<input type="checkbox"/>

Una vez agregados los agentes, debemos aplicar los cambios realizados en la sección de *Cola*, aplicar todos los cambios y esperar a que se complete la barra de progreso:



The screenshot shows the Pandora FMS interface. At the top, there is a green header with the text 'FRESH NEW POLICY - QUEUE' and a question mark icon. Below this, a success message box displays a green checkmark icon and the text 'SUCCESS Operation successfully added to the queue'. Underneath, there are links for 'Queue summary' and 'Queue filter'. A 'Total items : 1' label is positioned above a table. The table has columns for 'Policy', 'Agents', 'Operation', 'Progress', 'Finished', and 'Delete'. The first row contains 'Fresh new policy', 'All', 'Apply', a green progress bar, '-', and a trash icon. At the bottom right, there are three buttons: 'Refresh', 'Apply all', and 'Delete all'.

Policy	Agents	Operation	Progress	Finished	Delete
Fresh new policy	All	Apply	<div style="width: 100%; height: 10px; background-color: #90EE90;"></div>	-	

Una vez hecho, ya tenemos todos los módulos creados en la política desplegados a los agentes elegidos.

Las políticas nos permiten no sólo agregar módulos a grupos de agentes, también nos permiten incluir otro tipo de elementos como alertas, colecciones de archivos, plugins, etc. Además, cualquier modificación que hagamos en la política, como modificar el umbral de uno de sus módulos, será automáticamente heredado por todos los agentes incluidos en la política una vez que ésta sea aplicada.

Monitorización de agentes usando scripts personalizados

Esto es una forma *avanzada* de monitorizar grandes volúmenes de sistemas, parecidos entre sí, de una forma completamente "ad-hoc". Para ello tiene que disponer de herramientas que ya existen que le dan información de sus sistemas, algunos ejemplos pueden ser:

- Scripts que ya tenía que reportan información de sistemas remotos.
- Otros sistemas de monitorización ya funcionando que generan datos que se pueden reutilizar.
- Pequeños chequeos que son igual para un conjunto de XXX maquinas pero que no devuelven un unico dato sino varios simultáneamente. Si devolvieran datos de uno en uno, podría reutilizarlos como plugins para el servidor remoto.

La filosofía es simple: utiliza un script para generar las cabeceras de los XML de los agentes, poniendo el nombre de agente que quiera, y rellenando los datos de los módulos por un script, externo, que ejecutará como argumento. Este script externo debe generar datos correctos con el formato XML de Pandora (extremadamente sencillo!). El script principal cerrará el XML y lo moverá al path standard para procesar los ficheros de datos XML (`/var/spool/pandora/data_in`). Programe el script mediante CRON. Tiene más información sobre el formato XML que utiliza Pandora FMS para reportar los datos, consulte nuestros apéndices técnicos.

Script de agente remoto

Tiene un pequeño script en `/usr/share/pandora_server/util/pandora_remote_agent.sh` que admite

dos parámetros

```
-a <nombre de agente>
-f <fichero script que ejecutará>
```

De esta forma si tiene un script tal que `/tmp/sample_remote.sh` que contiene lo siguiente:

```
#!/bin/bash

PING=`ping 192.168.50.1 -c 1 | grep " 0% packet loss" | wc -l`

echo "<module>"
echo "<name>Status</name>"
echo "<type>generic_proc</type>"
echo "<data>$PING</data>"
echo "</module>"

ALIVE=`snmpget -Ot -v 1 -c artica06 192.168.70.100 DISMAN-EVENT-
MIB::sysUpTimeInstance | awk '{ print $3>=8640000 }'`

echo "<module>"
echo "<name>Alive_More_than_24Hr</name>"
echo "<type>generic_proc</type>"
echo "<data>$ALIVE</data>"
echo "</module>"

# Another script with returns XML
EXT_FILE=/tmp/myscript.sh

if [ -e "$EXT_FILE" ]
then
    $EXT_FILE
fi
```

Podrá generar un XML completo con el nombre de agente "agent_test" ejecutando el script de agente remoto de la siguiente manera:

```
/usr/share/pandora_server/util/pandora_remote_agent.sh -a agent_test -f
/tmp/sample_remote.sh
```

Supongamos que quiere ejecutar un mismo script contra XX maquinas, tendría que pasarle algunos datos, como usuario, IP, password al mismo script:

```
/usr/share/pandora_server/util/pandora_remote_agent.sh -a agent_test -f
"/tmp/sample_remote.sh 192.168.50.1"
```

Tendría que parametrizar el script `/tmp/sample_remote.sh` para coger los parámetros de línea de comandos y usarlos debidamente.

Programar el script mediante cron

Imagine que tiene 10 maquinas monitorizadas de esta manera:

```
/usr/share/pandora_server/util/pandora_remote_agent.sh -a agent_test1 -f
"/tmp/sample_remote.sh 192.168.50.1"
/usr/share/pandora_server/util/pandora_remote_agent.sh -a agent_test2 -f
"/tmp/sample_remote.sh 192.168.50.2"
/usr/share/pandora_server/util/pandora_remote_agent.sh -a agent_test3 -f
"/tmp/sample_remote.sh 192.168.50.3"
/usr/share/pandora_server/util/pandora_remote_agent.sh -a agent_test4 -f
"/tmp/sample_remote.sh 192.168.50.4"
/usr/share/pandora_server/util/pandora_remote_agent.sh -a agent_test5 -f
"/tmp/sample_remote.sh 192.168.50.5"
/usr/share/pandora_server/util/pandora_remote_agent.sh -a agent_test6 -f
"/tmp/sample_remote.sh 192.168.50.6"
/usr/share/pandora_server/util/pandora_remote_agent.sh -a agent_test7 -f
"/tmp/sample_remote.sh 192.168.50.7"
/usr/share/pandora_server/util/pandora_remote_agent.sh -a agent_test8 -f
"/tmp/sample_remote.sh 192.168.50.8"
/usr/share/pandora_server/util/pandora_remote_agent.sh -a agent_test9 -f
"/tmp/sample_remote.sh 192.168.50.9"
/usr/share/pandora_server/util/pandora_remote_agent.sh -a agent_test10 -f
"/tmp/sample_remote.sh 192.168.50.10"
```

Meta todas estas líneas en un nuevo script, p.e: "/tmp/my_remote_mon.sh" y dele permisos de ejecución, y añada la siguiente linea al crontab de root:

1. */5 * * * * root /tmp/my_remote_mon.sh

Esto hará que ese script se ejecute en el sistema cada 5 minutos. Puede ir añadiendo máquinas al script.

Si quieres saber más información sobre la monitorización de sistemas, sus ventajas y el proceso a seguir para hacer una correcta monitorización consulta nuestro [artículo de monitorización de sistemas](#).

[Volver al índice de documentación de Pandora FMS](#)