



Componentes de Wizard



h:
<https://pandorafms.com/manual/!775/>
manent link:
https://pandorafms.com/manual/!775/es/documentation/pandorafms/technical_annexes/25_pfms_new_wizard_component
4/03/18 21:03



Componentes de Wizard

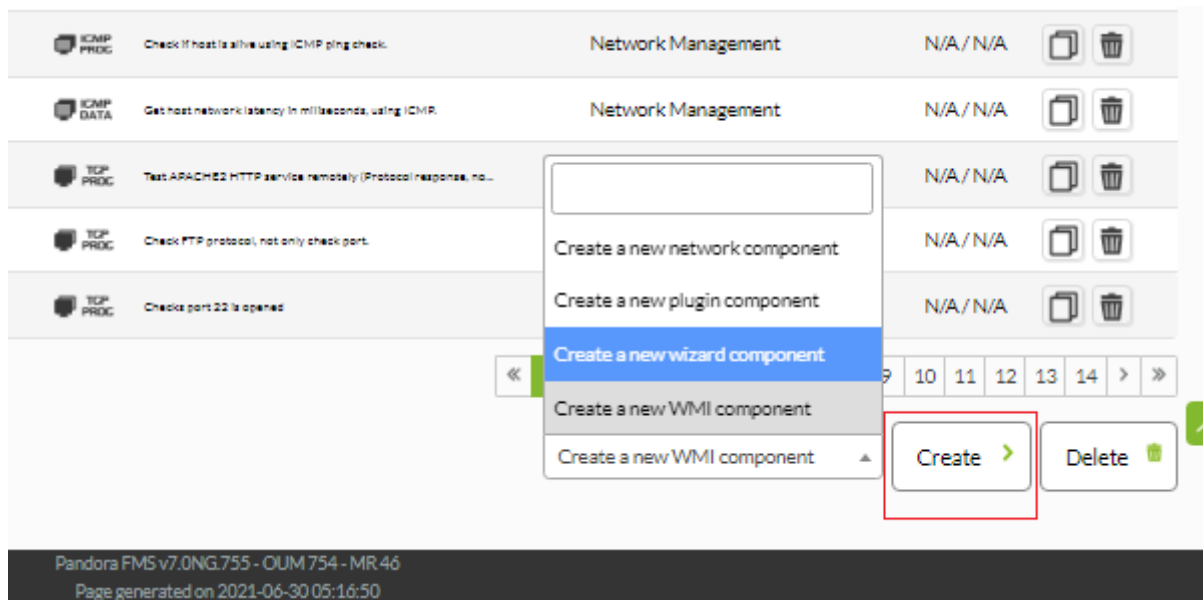
Dentro de las capacidades del wizard SNMP y el wizard WMI, se encuentran un tipo de componentes remotos llamados *Componentes de wizard*.

Estos componentes permiten establecer una configuración base para los módulos que serán generados en los agentes al ejecutar cualquiera de los wizard (SNMP o WMI). Además ofrece la posibilidad de que con un solo componente se puedan generar varios módulos de forma dinámica. Por ejemplo, un componente para escanear las distintas unidades de almacenamiento de un dispositivo o los procesos en ejecución.

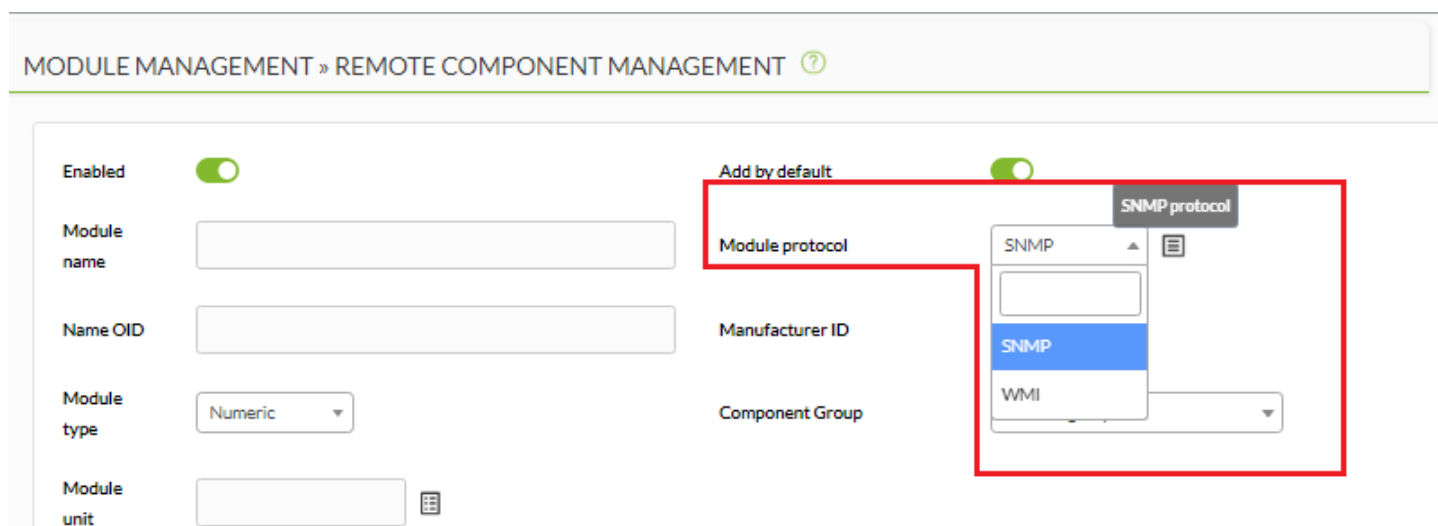
Se pueden crear estos componentes desde el menú Configuration → Templates → Remote components,

The screenshot displays the Pandora FMS Enterprise web interface. The top left features the Pandora FMS logo and the text "the Flexible Monitoring System". A search bar at the top right contains the text "Net" and a "Go to" button with a wrench icon, which is highlighted by a red box. Below the search bar, the breadcrumb "MODULE MANAGEMENT » REMOTE COMPONENT MANA" is visible. The left sidebar contains a navigation menu with categories: Monitoring, Topology maps, Reporting, Events, Workspace, Tools, Discovery, Resources, Profiles, Configuration (highlighted), Alerts, Events, Servers, Setup, Admin tools, Links, Update manager, and Module library. The main content area shows a "Group" dropdown set to "All" and a "Free Search" field. A secondary menu is open over the "Configuration" section, listing options like "Module templates", "Private Enterprise Numbers", "Local components", and "Remote components" (which is highlighted). Below this, a list of components is shown, including "OS Users" with a checkbox, "CPU Load in Operating System (UNIX MIB)", and "Active users in Operating System (UNIX MIB)". The footer contains the version information "Pandora FMS v7.0NG.755 - OUM 754 - MR 46" and the generation timestamp "Page generated on 2021-06-25 04:00:50".

Seleccione la opción Create a new wizard component y pulse el botón Create.



En el cuadro de diálogo de creación existe una opción de vital importancia: el protocolo de Módulo a utilizar. Existen dos protocolos disponibles, SNMP (por defecto) y WMI, y aunque ambos comparten campos comunes también difieren en otros campos.



Los campos comunes son:

- Enabled: Al activar este *token* se indica que el componente se intentará escanear al lanzar el *wizard*.
- Add by default: Permite elegir si los módulos generados por el componente estarán marcados para agregarse por defecto al lanzar el *wizard*. Es decir, si se activa el *token*, los módulos generados por el componente estarán marcados por defecto en una vista que encontraremos más adelante y se añadirán al Agente. Luego en esa vista se pueden hacer modificaciones y desmarcar o marcar a voluntad y cambiar umbrales, descripciones, etc.
- Module name: Nombre que tendrá el componente, y nombre por defecto para los módulos generados por el mismo. Se podrá hacer uso de algunas macros que se explican en las siguientes secciones.
- Module protocol: permite elegir entre SNMP y WMI (cambian algunos campos).
- Module type: En este desplegable se puede escoger el tipo de dato que obtendrán los módulos generados por el componente.
- Component group: Grupo al que pertenecerá el componente. Permite organizar la forma en que se presentarán los módulos.
- Module unit: Unidad de los datos obtenidos por los módulos generados por el componente. Es un

campo totalmente editable, por lo que puede agregar la medida que necesite.

- **Warning status:** En esta sección se puede establecer un umbral por defecto para el estado `warning` de los módulos generados por el *wizard*. A pesar de indicar aquí un rango, habrá posibilidad de personalizarlo para cada módulo en la vista final que recoge todos los módulos encontrados.
- **Critical status:** En esta sección puede establecer un umbral por defecto para el estado `critical` de los módulos generados por el *wizard*. A pesar de indicar aquí un rango, habrá posibilidad de personalizarlo para cada módulo en la vista final que recoge todos los módulos encontrados.
- **Description:** Descripción que tendrá el componente y a su vez los módulos generados por el mismo. Se podrá hacer uso de algunas macros.
- **Scan type:** Permite elegir entre dos modalidades de exploración que podrán realizar los *wizards* con este componente. Este campo determina si un componente generará un solo módulo o varios. El valor seleccionado afectará a como se deben rellenar otros campos específicos de cada *wizard*.
 - **Fixed:** El componente solo generará un módulo. Por ejemplo, obtener el *uptime* (tiempo funcionando) del dispositivo por SNMP.
 - **Dynamic:** El componente podrá generar uno o más módulos. Por ejemplo, obtener el porcentaje de uso de las unidades de disco por WMI.
- **Execution type:** Con este campo se indica el tipo de ejecución para los módulos generados por el componente. Sirve para determinar el servidor de Pandora FMS al que pertenecerán los módulos en el momento de su creación dependiendo desde dónde se ejecute el *wizard*.
 - **Network:** Los módulos generados por el componente obtendrán sus datos con los mecanismos propios de Pandora FMS para módulos SNMP y WMI. Estos son: los **servidores de red**, **servidor WMI** y **servidor satélite**.
 - **Plugin:** Los módulos generados por el componente obtendrán sus datos de la ejecución de comandos, *plugins* o *scripts* personalizados. De esta manera, serán ejecutados por el **servidor de plugins** o por el servidor satélite mediante módulos de tipo `exec`.

SNMP wizard

MODULE MANAGEMENT » REMOTE COMPONENT MANAGEMENT ?

Enabled Add by default

Module name Module protocol

Name OID Manufacturer ID

Module type Component Group

Module unit

Warning Min. Max. Critical Min. Max.

Inverse interval Inverse interval

Description

Scan Type Execution type

Value OID

Go back ✖ Create ✖

Pandora FMS v7.0NG.755 - OUM 754 - MR 46
Page generated on 2021-06-30 05:23:29

Los campos específicos para componentes del wizard SNMP son:

- Name OID
- Manufacturer ID
- Cuando Execution type está establecido en Network:
 - Value OID
- Cuando Execution type está establecido en Plugin:
 - Macros OID
 - Value operation
 - Satellite execution
 - Server plugin

MODULE MANAGEMENT » REMOTE COMPONENT MANAGEMENT ?

Enabled Add by default

Module name

Module protocol


SNMP 

Name OID

Manufacturer ID

All 


Module type

Boolean 

Component Group

General group 

Module unit

Warning

Min. Min. Max.

Critical

Max. Inverse interval Inverse interval

Description

Scan Type

Fixed 

Execution type

Plugin 

Macros OID



_oid_1_ Value operation Satellite execution

Server plugin

IPMI Plugin 

Plugin to get IPMI monitors from a IPMI

Device.

Target IP Password Username Additional Options Sensor Go back Create 

Permite indicar un OID del cual se obtendrá un valor que podrá ser añadido al nombre del módulo mediante una macro. Resulta especialmente útil cuando se obtienen múltiples módulos generados por un componente dinámico. De esta manera se consigue que tengan nombres distintos por defecto. Pero no se limita a los componentes dinámicos, ya que se puede utilizar también para los componentes de escaneo fijo.

El valor de este OID es almacenado en la macro `_nameOID_`, que puede ser usada en el campo `Module name`.

Si se utiliza en componentes dinámicos, el OID indicado en este campo debe ser una rama de SNMP y no un OID final. Por ejemplo, si se indica el OID `.1.3.6.1.4.1.2021.10.1.2`, los valores que tendrá la macro en cada módulo se obtendrán de los OID `.1.3.6.1.4.1.2021.10.1.2.x`, donde `x` representa cada una de las terminaciones que puede tener la rama.

Si se utiliza en componentes fijos, el OID indicado en este campo debe ser un OID final. Por ejemplo, si se indica el OID `.1.3.6.1.2.1.1.5.0`, el valor que tendrá la macro en el módulo se obtendrá directamente de ese OID.

Manufacturer ID

Permite indicar el ID de un fabricante específico para el cual el componente del *wizard* SNMP tendrá efecto. De esta manera, para todos los dispositivos contra los que se lance el *wizard*, y cuyo Private Enterprise Number (PEN) esté registrado en Pandora FMS para el ID del fabricante asignado al componente, se intentarán obtener los módulos generados por el mismo. Por ejemplo, un componente asignado a `general_snmp`, se escaneará para todos los dispositivos con los PEN 2021 y 8072.

Si se indica como fabricante All, el componente se escaneará para cualquier PEN registrado en Pandora FMS.

Debe estar registrado el Private Enterprise Number (PEN) en la consola de Pandora FMS para el uso de Manufacturer ID

SNMP Ejecución tipo Network

Cuando el tipo de ejecución sea Network:

Value OID:

Permite indicar el OID a partir del cual se obtendrán los datos de los módulos generados por el componente. Si se utiliza en componentes dinámicos, el OID indicado en este campo debe ser una

rama de SNMP y no un OID final. Por ejemplo, si se indica el OID `.1.3.6.1.4.1.2021.10.1.3`, los valores que tendrán los módulos se obtendrán de los OIDs `.1.3.6.1.4.1.2021.10.1.3.x`. Además, el nodo `x` de cada OID debe tener el mismo valor para el nodo `X` del campo Name OID, en el caso de utilizarse.

Si se utiliza en componentes fijos, el OID indicado en este campo debe ser un OID final. Por ejemplo, si se indica el OID `.1.3.6.1.4.1.2021.11.9.0`, el valor que tendrá el módulo se obtendrá directamente de ese OID.

Scan Type	Fixed ▼	Execution type	Network ▼
Value OID	1.3.6.1.4.1.2021.11.9.0		

SNMP ejecución tipo Plugin

Cuando el tipo de ejecución sea Plugin:

Macros OID → `_oid_N_`

El propósito principal de utilizar componentes de tipo *plugin* es poder realizar operaciones con los valores de uno o más OID en el mismo dispositivo, como por ejemplo obtener el porcentaje de memoria usada a partir de los bytes de memoria usada y los bytes totales de memoria disponibles.

Es por ello que en estos componentes se pueden indicar tantos OID como se necesiten para utilizarlos en otros campos.

Además, estos OID, o sus valores, se podrán utilizar a partir de las macros `_oid_N_`. Dependiendo de en cuál de los campos siguientes se utilice la macro, se usará el valor del OID o el OID en sí mismo.

Si se utilizan en componentes dinámicos, los OID indicados en estos campos deben ser una rama de SNMP y no un OID final. Por ejemplo, si se indica el OID `.1.3.6.1.4.1.3375.2.1.7.4.2.1.3`, los valores que tendrán los módulos se obtendrán de los OIDs `.1.3.6.1.4.1.3375.2.1.7.4.2.1.3.x`. Además el nodo `X` de cada OID debe tener el mismo valor para el nodo `X` del resto de OID usados y del campo Name OID, en el caso de utilizarse.

Si se utiliza en componentes fijos, los OID indicados en estos campos deben ser un OID final. Por ejemplo, si se indica el OID `.1.3.6.1.4.1.2021.4.6.0`, el valor que tendrá el módulo se obtendrá directamente de ese OID.

Value operation

Permite indicar una operación aritmética con la cual se obtendrá el valor actual de cada módulo generado por el componente en la vista previa de los módulos del *wizard*. En ningún caso afecta para la ejecución definitiva de los módulos generados.

Acepta los caracteres + - * / () . , números y las macros `_oid_N_` de las cuales se obtendrán los valores para la operación. Por ejemplo:

```
(_oid_1_ * 100) / _oid_2_
```

Satellite execution

Ofrece la posibilidad de indicar la ejecución que deba realizar un Servidor Satélite para los módulos generados cuando el *wizard* se lance desde un Servidor Satélite mediante el uso del `exec server`. Se trata del comando, *plugin* o *script* que se deba usar en un `module_exec` de servidor satélite.

Acepta el uso de las macros para el wizard SNMP (se detallarán más adelante) y de las macros `_oid_N_` para obtener los OID usados en cada módulo.

El Servidor Satélite distribuye una serie de plugins recomendados para estos componentes:

- /etc/pandora/satellite_plugins/wizard_snmp_module
- /etc/pandora/satellite_plugins/wizard_snmp_process

Server plugin

The screenshot shows a configuration form for Pandora FMS. A dropdown menu is open, displaying a list of plugins. The 'Server plugin' field is set to 'IPMI Plugin'. The 'Target IP' field is also set to 'IPMI Plugin'. The 'Password' field is set to 'UDP port check'. The 'Additional Options' field is set to 'SNMP remote'. The 'Username' field is empty. The 'Sensor' field is empty. The dropdown menu lists the following options: IPMI Plugin, DNS Plugin, UDP port check, SMTP Check, MySQL Plugin, SNMP remote, Packet Loss, Wizard SNMP module, Wizard SNMP process, Wizard WMI module, Network bandwidth, and SNMP. The footer of the page reads: Pandora FMS v7.0NG.755 - OUM 754 - MR 46, Page generated on 2021-06-30 14:16:45.

Mediante este desplegable se podrá indicar un *plugin* registrado previamente en la Consola de Pandora FMS, el cual será utilizado por el servidor de plugins con cada módulo generado por el componente. La elección de un *plugin* muestra a su vez en el formulario los campos específicos para la utilización del mismo.

Los campos propios del plugin aceptan el uso de las macros para el wizard SNMP y de las macros `_oid_N_` para obtener los OIDs usados en cada módulo.

La consola de Pandora FMS cuenta con una serie de plugins ya registrados recomendados para estos componentes:

- Wizard SNMP module.
- Wizard SNMP process.

Por ejemplo, al seleccionar Wizard SNMP module obtendrá los siguientes campos a rellenar:

Scan Type	Fixed ▼	Execution type	Plugin ▼
Macros OID			
_oid_1	<input type="text" value="1.3.6.1.4.1.2021.4.6.0"/>		
_oid_2	<input type="text" value="1.3.6.1.4.1.2021.4.5.0"/>		
⊕ 🗑			
Value operation	<input type="text" value="(_oid_1 * 100) / _oid_2"/>		
Satellite execution	<input "_cc"="" "_port_"="" "_version_"="" -community="" -port="" -version="" _address_"="" type="text" value="/etc/pandora/satellite_plugins/wizard_snmp_module -host "/>		
Server plugin	Wizard SNMP module ▼	Get the result of an arithmetic operation using several OIDs values.	
Port	<input type="text" value="_port_"/>	Host	<input type="text" value="_address_"/>
Community	<input type="text" value="_community_"/>	Version	<input type="text" value="_version_"/>
Username (v3)	<input type="text" value="_auth_user_"/>	Security level (v3)	<input type="text" value="_sec_level_"/>
Authentication password (v3)	<input type="text" value="_auth_pass_"/>	Authentication method (v3)	<input type="text" value="_auth_method_"/>
Privacy password (v3)	<input type="text" value="_priv_pass_"/>	Privacy method (v3)	<input type="text" value="_priv_method_"/>
Operation	<input type="text" value="(_o1_ * 100) / _o2_"/>	OID list	<input type="text" value="_oid_1_ _oid_2_"/>

Las macros específicas para los componentes del wizard SNMP que se pueden usar en los campos de ejecuciones de tipo *plugin* son:

- **_address_** : Dirección IP usada en el wizard SNMP. No se sustituirá esta macro cuando el *wizard* se lance en una política.
- **_port_** : Puerto usado en el wizard SNMP.
- **_version_** : Versión de SNMP usada en el wizard SNMP. Puede tener los valores 1, 2c o 3.
- **_community_** : Comunidad SNMP usada en el wizard SNMP.
- **_sec_level_** : Nivel de seguridad de SNMPv3 usado en el wizard SNMP. Puede tener los valores noAuthNoPriv, authNoPriv o authPriv
- **_auth_user_** : Usuario de SNMPv3 usado en el wizard SNMP.
- **_auth_method_** : Método de autenticación de SNMPv3 usado en el wizard SNMP. Puede tener los valores MD5 o SHA.
- **_auth_pass_** : Contraseña de autenticación de SNMPv3 usada en el wizard SNMP.
- **_priv_method_** : Método de privacidad de SNMPv3 usado en el wizard SNMP. Puede tener los valores DES o AES.
- **_priv_pass_** : Contraseña de privacidad de SNMPv3 usada en el wizard SNMP.

WMI wizard

MODULE MANAGEMENT » REMOTE COMPONENT MANAGEMENT ?

Enabled

Module name

Module type Boolean

Module unit

Warning Min. 0 Max. 0

Inverse interval

Description

Scan Type Fixed

Add by default

Module protocol WMI

Component Group General group

Critical Min. 0 Max. 0

Inverse interval

Execution type Network

WMI class

Query key field (_field_wmi_0_)

Query extra fields

_field_wmi_1_

Query filters + 🗑

Scan

Execution

Field value Key string

Go back ✖ Create ✖

Pandora FMS v7.0NG.755 - OUM 754 - MR.46
Page generated on 2021-06-30 05:23:29

WMI ejecución tipo Network

Cuando el tipo de ejecución sea Network:

WMI class

Permite especificar la clase WMI a consultar. Las clases WMI son predefinidas que se incluyen en cada espacio de nombres del núcleo de Windows Management Instrumentation® (WMI).

Query key field (`_field_wmi_0_`)

Query extra fields (`_field_wmi_1_`)

Query filters → Execution:

Este espacio permite indicar las condiciones para la consulta WMI lanzada por cada módulo generado por el componente. Acepta el uso de las macros con los nombres de los campos de la consulta (`_FIELDNAME_`) para obtener el valor de cada registro en ese campo. Por ejemplo:

```
DriveType = 3 AND DeviceID = '_DeviceID_'.
```

La consulta de ejecución final de un módulo generado por el componente para obtener el espacio libre de la unidad de disco C : es:

```
SELECT DeviceID, FreeSpace FROM Win32_LogicalDisk WHERE DriveType = 3 AND DeviceID = 'C:'
```

Field value

Indicaremos el número del campo de la consulta WMI del cual se quiera obtener el valor del módulo, siendo el campo 0 el campo clave de la clase y los campos 1, y superiores, los adicionales de la clase.

Key string

Permite convertir el valor del módulo en booleano (1 o 0) dependiendo de si el valor del campo indicado en *Field value* coincide con la cadena de texto indicada en este campo.

La opción Key string no se tendrá en cuenta cuando el wizard se lance desde un Servidor Satélite mediante `exec server`

Scan Type	Dynamic ▾	Execution type	Network ▾
WMI class	Win32_Processor		
Query key field (_field_wmi_0_)	DeviceID		
Query extra fields			
_field_wmi_1_	LoadPercentage		
	⊕ 🗑		
Query filters			
Scan			
Execution	DeviceID = '_DeviceID_'		
Field value	1	Key string	

WMI ejecución tipo Plugin

Scan Type	Dynamic ▾	Execution type	Plugin ▾
WMI class			
Query key field (_field_wmi_0_)			
Query extra fields			
_field_wmi_1_			
	⊕ 🗑		
Query filters			
Scan			
Value operation			
Satellite execution			
Server plugin	IPMI Plugin ▾	Plugin to get IPMI monitors from a	
	IPMI Device.		
Target IP			
Password		Username	
Additional Options		Sensor	

Cuando el tipo de ejecución sea Plugin:

Comparte campos comunes con la ejecución Network hasta el campo Scan (Query filters).

Value operation

El propósito principal de utilizar componentes de tipo *plugin* es poder realizar operaciones con los valores de distintos campos de la consulta, como por ejemplo obtener el porcentaje de disco usado a partir de los bytes de disco libre y los bytes totales de disco disponibles.

Este campo permite indicar una operación aritmética con la cual se obtendrá el valor actual de cada módulo generado por el componente en la vista previa de los módulos del *wizard*. En ningún caso afecta para la ejecución definitiva de los módulos generados.

Acepta los caracteres + - * / () . , números y las macros con los nombres de los campos de la clase (`_FIELDNAME_`), de las cuales se obtendrán los valores para la operación. Por ejemplo:

```
((_Size_ - _FreeSpace_) * 100) / _Size_
```

Satellite execution

Permite indicar la ejecución que deba realizar un Servidor Satélite para los módulos generados cuando el *wizard* se lance desde un Servidor Satélite mediante el uso del *exec server*. Se trata del comando, *plugin* o *script* que se deba usar en un `module_exec` de Servidor Satélite.

Acepta el uso de las macros para el wizard WMI y de las macros `_class_wmi_` para obtener el nombre de la clase WMI y `_field_wmi_N_` para obtener los nombres de los campos de la clase usados en cada módulo.

El Servidor Satélite distribuye un plugin recomendado para estos componentes:

```
/etc/pandora/satellite_plugins/wizard_wmi_module
```

Server plugin

Permite indicar un plugin registrado en la consola de Pandora que será utilizado por el servidor de plugins con cada módulo generado por el componente. La elección de un plugin muestra a su vez en el formulario los campos específicos para la utilización del mismo.

Los campos propios del plugin aceptan el uso de las macros para el wizard WMI y de las macros `_class_wmi_` para obtener el nombre de la clase WMI y `_field_wmi_N_` para obtener los nombres de los campos de la clase usados en cada módulo.

La Consola de Pandora FMS cuenta con un *plugin* ya registrado recomendado para estos componentes: Wizard WMI module.

Scan Type	<input type="text" value="Dynamic"/>	Execution type	<input type="text" value="Plugin"/>
WMI class	<input type="text" value="Win32_LogicalDisk"/>		
Query key field (_field_wmi_0_)	<input type="text" value="DeviceID"/>		
Query extra fields			
_ field_wmi_1_	<input type="text" value="Size"/>		
_ field_wmi_2_	<input type="text" value="FreeSpace"/>		
	<input type="button" value="+"/> <input type="button" value="🗑"/>		
Query filters			
Scan	<input type="text" value="DriveType = 3"/>		
Value operation	<input type="text" value="((_Size_ - _FreeSpace_) * 100) / _Size_"/>		
Satellite execution	<input "_namespac"="" -namespace="" _address_"="" type="text" value="/etc/pandora/satellite_plugins/wizard_wmi_module -host "/>		
Server plugin	<input type="text" value="Wizard WMI module"/> <input type="button" value="v"/> Get the result of an arithmetic operation using distinct fields in a WMI quer...		
Namespace (Optional)	<input type="text" value="_namespace_wmi_"/>	Host	<input type="text" value="_address_"/>
Password	<input type="text" value="_pass_wmi_"/>	User	<input type="text" value="_user_wmi_"/>
Fields list	<input type="text" value="_field_wmi_1_ _field_wmi_2_"/>	WMI Class	<input type="text" value="_class_wmi_"/>
Operation	<input type="text" value="((_f1_ - _f2_) * 100) / _f1_"/>	Query filter (Optional)	<input type="text" value="DeviceID = '_DeviceID_'"/>

Las macros específicas para los componentes del wizard WMI que se pueden usar en los campos de ejecuciones de tipo *plugin* son:

- *_address_* : Dirección IP usada en el wizard WMI. No se sustituirá esta macro cuando el *wizard* se lance en una política.
- *_namespace_wmi_* : *Namespace* usado en el wizard WMI.
- *_user_wmi_* : Usuario utilizado en el wizard WMI.
- *_pass_wmi_* : Contraseña utilizada en el wizard WMI.

Los campos específicos para componentes del wizard WMI son:

* ****WMI class****: Hace referencia a la clase WMI que se utilizará en las consultas de los módulos generados por el componente. Por ejemplo, `//Win32_LogicalDisk//`.

Se puede utilizar en otros campos de este mismo formulario mediante la macro `_class_wmi_`.

Query key field (`_field_wmi_0_`)

Es el nombre del campo clave que se obtendrá en la consulta WMI usada en los módulos generados. Por lo general las clases WMI tienen un campo clave que siempre devuelven en cualquier consulta tanto si se indica como si no. Ese sería el campo que se debería indicar aquí. Por ejemplo, el campo clave de la clase `Win32_Processor` sería `DeviceID`.

El nombre de este campo se puede obtener en otros campos del formulario mediante la macro `_field_wmi_0_`, y el valor que tenga el campo para cada registro de la consulta WMI se puede obtener mediante una macro con el mismo nombre del campo (`_FIELDNAME_`). Estas macros `_FIELDNAME_` se pueden utilizar, entre otros, en los campos Module name y Description del componente, para generar nombres y descripciones de forma dinámica. Por ejemplo, para el campo `DeviceID` la macro con el valor sería `_DeviceID_`.

Query extra fields → `_field_wmi_N_`:

En estos campos indicaremos los nombres de los campos adicionales que se deban utilizar en la consulta WMI usada en los módulos generados.



Los nombres de estos campos se pueden obtener en otros campos del formulario mediante las macros `_field_wmi_N_`, y los valores que tengan los campos para cada registro de la consulta WMI se pueden obtener mediante macros con los mismos nombres de los campos (`_FIELDNAME_`). Estas macros `_FIELDNAME_` se pueden utilizar, entre otros, en los campos Module name y Description del componente, para generar nombres y descripciones de forma dinámica. Por ejemplo, para el campo `FreeSpace` la macro con el valor sería `_FreeSpace_`.

Query filters → Scan

En este espacio se indican las condiciones para la consulta WMI lanzada en el escaneo, la cual permitirá obtener uno o más registros. Por ejemplo: `DriveType = 3`.

En los componentes del wizard WMI, se generará un módulo distinto por cada registro devuelto por la consulta de escaneo WMI. Basándonos en los ejemplos aportados hasta ahora, la consulta de escaneo que se realizaría obtendría el espacio libre de las unidades de disco del equipo Windows:

```
SELECT DeviceID, FreeSpace FROM Win32_LogicalDisk WHERE DriveType = 3
```

Scan Type	Dynamic ▾	Execution type	Plugin ▾
WMI class	Win32_LogicalDisk		
Query key field (_field_wmi_0_)	DeviceID		
Query extra fields			
_field_wmi_1_	Size		
_field_wmi_2_	FreeSpace		
	 		
Query filters			
Scan	DriveType = 3		

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