

PANDORAFMS
E N T E R P R I S E

Pandora FMS
Manual Administrador
Monitorización Sistemas Solaris



Manual Administrador Monitorización Sistemas Solaris

© Artica Soluciones Tecnológicas 2005-2012

Índice de contenido

1	Histórico de cambios.....	3
2	Introducción.....	4
3	Matriz de compatibilidad	5
4	Documentación a entregar por el Área que requiere la monitorización.....	6
5	Módulos de monitorización.....	7
6	Definición del los módulos de monitorización.....	8
6.1	Módulos generales.....	8
6.1.1	CPU User.....	8
6.1.2	CPU System.....	8
6.1.3	Process Running.....	9
6.1.4	Physical Free Memory.....	9
6.1.5	Used space	9
6.1.6	Number processes.....	10
6.1.7	Zombie processes.....	10
6.1.8	Process memory.....	10
6.1.9	Process CPU.....	11
6.1.10	NTPD Instances.....	11

1 HISTÓRICO DE CAMBIOS

Fecha	Autor	Cambio	Versión
17/03/12	Dario	Primera versión del plugin	v1r1

2 INTRODUCCIÓN

Este documento tiene como objetivo la descripción de la monitorización de cualquier máquina con sistema Operativo Solaris.

Se han elegido una serie de módulos en base a nuestra experiencia en monitorización de sistemas y las necesidades de algunos de nuestros clientes.

Para la extracción de la información se utiliza la capa de recolección para los módulos de Pandora FMS susceptibles de ello, reduciendo así la carga de los sistemas a monitorizar. O se han añadido mediante *module_exec* si es un comando o mediante *module_plugin* si hablamos de un plugin de agente.

3 MATRIZ DE COMPATIBILIDAD

La matriz de compatibilidad para el plugin se muestra a continuación:

Sistemas donde se ha probado	<ul style="list-style-type: none"> • SunOs 5.x
-------------------------------------	---

Sistemas donde debería funcionar	<ul style="list-style-type: none"> • Mismo sistema o superior
---	--

4 DOCUMENTACIÓN A ENTREGAR POR EL ÁREA QUE REQUIERE LA MONITORIZACIÓN.

La información que debe entregar el área que requiere la monitorización es la siguiente:

- El usuario que ejecute el agente de Pandora FMS debe tener los suficientes permisos para ejecutar los comandos y acceder a los ficheros necesarios sin problemas.
- Requisitos para que funcione correctamente esta monitorización son los siguientes:
 - Agente Pandora FMS instalado
- Información para los módulos:
 - Servicios a monitorizar
 - Discos a monitorizar

5 MÓDULOS DE MONITORIZACIÓN

- **CPU User:** Este módulo devuelve el porcentaje de CPU ocupado por procesos de usuario
- **CPU System:** Este módulo devuelve el porcentaje de CPU ocupado por procesos del sistema
- **Process Running:** El módulo busca si un determinado proceso está corriendo
- **Physical Free Memory:** El módulo devuelve el porcentaje de memoria física libre
- **Used space:** El módulo devuelve el porcentaje de directorio ocupado
- **Number processes:** Devuelve el número de procesos que están corriendo en la máquina.
- **Zombie processes:** Devuelve el número de procesos zombies
- **Process memory:** El módulo devuelve el porcentaje de memoria en uso por un proceso
- **Process CPU:** El módulo devuelve el porcentaje de CPU en uso por un proceso
- **NTPD Instances:** El módulo el número de instancias del proceso ntpd corriendo

6 DEFINICIÓN DEL LOS MÓDULOS DE MONITORIZACIÓN

A continuación se describen los diferentes módulos de monitorización, así como el código de cada uno de ellos.

6.1. Módulos generales

6.1.1. *CPU User*

Este módulo devuelve el porcentaje de CPU ocupado por procesos de usuario. La sintaxis para el módulo sería la siguiente:

```
module_begin
module_name CPU User
module_type generic_data
module_exec vmstat 1 2 | tail -1 | awk '{ print $20 }' | tr -d "\n"
module_max 100
module_min 0
module_description CPU (%) used by user
module_end
```

6.1.2. *CPU System*

Este módulo devuelve el porcentaje de CPU ocupado por procesos del sistema. La sintaxis para el módulo sería la siguiente:

```
module_begin
module_name CPU System
module_type generic_data
module_cpuusage all
module_max 100
module_min 0
module_description CPU (%) used by system
module_end
```


6.1.3. *Process Running*

El módulo busca si un determinado proceso está corriendo. La sintaxis del módulo es la siguiente:

```
module_begin
module_name Process Running xxx
module_type generic_proc
module_service xxx
module_description Proces xxx Running
module_end
```

Sustituiremos xxx por el proceso concreto que queramos monitorizar.

6.1.4. *Physical Free Memory*

El módulo devuelve el porcentaje de memoria física libre. La sintaxis del módulo sería la siguiente:

```
module_begin
module_name Physical Free Memory
module_type generic_data
module_freepcentmemory
module_description Porcentaje Memoria física libre
module_end
```

6.1.5. *Used space*

El módulo devuelve el porcentaje de directorio ocupado, la sistaxis es la siguiente:

```
module_begin
module_name Used space xxx
module_type generic_data
module_occupiedpercentdisk /xxx
module_description % Directorio ocupado
module_end
```

Cambiaremos xxx por el volumen a monitorizar.

6.1.6. *Number processes*

Devuelve el número de procesos que están corriendo en la máquina. La sintaxis del módulo es la siguiente:

```
module_begin
module_name Number processes
module_type generic_data
module_exec ps -e | wc -l | tr -d "\n" | tr -d " "
module_description Number processes
module_end
```

6.1.7. *Zombie processes*

Devuelve el número de procesos zombies, la sintaxis es la siguiente:

```
module_begin
module_name Zombie processes
module_type generic_data
module_exec ps -el | awk '$2~/\"Z\"/{print $2}' | wc -l | tr -d "\n" | tr -d " "
module_description Zombie processes
module_end
```

6.1.8. *Process memory*

El módulo devuelve el porcentaje de memoria en uso por un proceso, la sintaxis es la siguiente:

```
module_begin
module_name Process memory xxx
module_type generic_data
module_memproc xxx
module_description % used memory of process
module_end
```

Cambiaremos xxx por el proceso a monitorizar.

6.1.9. *Process CPU*

El módulo devuelve el porcentaje de CPU en uso por un proceso, la sintaxis es la siguiente:

```
module_begin
module_name Process CPU xxx
module_type generic_data
module_cpuproc xxx
module_description % CPU used by process
module_end
```

Cambiaremos xxx por el proceso a monitorizar.

6.1.10. *NTPD Instances*

El módulo el número de instancias del proceso ntpd corriendo.

```
module_begin
module_name NTPD Instances
module_type generic_data
module_exec ps -Af | grep ntpd | grep -v "grep" | wc -l | tr -d " " | tr -d "\n"
module_description Instancias NTPD en ejecución
module_end
```