

### Configuración de SNMP

Inicialmente se requieren de tres archivos de configuración:

1. snmp.conf
2. snmpd.conf
3. snmptrapd.conf

Para generar el archivo snmpd.conf de una manera mas sencilla, se puede usar la interfaz escrita en perl, llamada snmpconf.

La configuración también se puede realizar a mano siguiendo los siguientes pasos y haciendo uso de otras instrucciones abajo descritas.

Pero antes de empezar es necesario tener en cuenta que SNMP utiliza restricciones de acceso a un elemento de la red por medio de nombres de comunidades y tienen las siguientes características.

- ✓ Se usan para propósitos de control administrativo, prácticamente se utilizan como si fueran passwords para el control de acceso.
- ✓ Permite identificar el tipo de relación existente entre “el que habla” y “el que escucha”, para identificar que datos pueden ser accesados.
- ✓ Son incluidas en cada petición o respuesta, dentro de una comunicación entre dispositivos que usan SNMP.
- ✓ El nombre de comunidad que la mayoría de ordenadores utilizan para manejo de información pública es public y generalmente son para datos estadísticos

#### Paso 1.

Identificar un nombre de comunidad como un nombre de seguridad para mensajes del tipo get (estadísticas de solo lectura) y set (actualizaciones – lectura/escritura). También se establecen las IP's desde las cuales se enviarán peticiones (0.0.0.0 significa todas). Por ejemplo:

#	<u>nombre de seguridad</u>	<u>origen de consulta</u>	<u>nombre de comunidad</u>
com2sec	local	localhost	mimaquina
com2sec	mired	148.204.0.0/16	micomunidad

#### Paso 2.

Relacionar los nombres de seguridad dentro de nombres de grupos. Por ejemplo:

#	<u>nombre de grupo</u>	<u>modelo de seguridad</u>	<u>nombre de seguridad</u>
group	MiRWGroup	v1	local
group	MiRWGroup	v2c	local
group	MiRGroup	v1	mired
group	MiRGroup	v2c	mired

#### Paso 3.

## Configuración de Simple Network Management Protocol

---

Crear una vista para dar derechos a cada grupo. Por ejemplo:

#	<u>permiso</u>	<u>incl/excl</u>	<u>sub-arbol</u>
view	all	included	.1

### Paso 4.

Dar permisos a los grupos para acceder a las vistas. Por ejemplo:

#	<u>nombre de grupo</u>	<u>contexto</u>	<u>modelo seg.</u>	<u>nivel</u>	<u>match</u>	<u>read</u>	<u>write</u>	<u>notif</u>
access	MiRGroup	""	any	noauth	exact	all	none	none
access	MiRWGroup	""	any	noauth	exact	all	all	none

### Paso 5.

Configurar el sistema de contacto para información, a través de un correo electrónico y la ruta de donde se encuentra el archivo de configuración de snmp (snmpd.conf). Por ejemplo:

```
syslocation /etc/snmp/snmpd.conf
syscontact Me <usuario@sitio.com>
```

Cabe destacar que todos los ejemplos que se han presentado van de la mano y son efectivos, sin embargo, la seguridad que proporcionen depende de las necesidades de cada empresa.

Estos son los pasos esenciales para configurar el servidor de snmp, con esto se pueden hacer consultas vía snmp para realizar pruebas con la misma máquina y una vez familiarizado con los comandos, es conveniente utilizarlo con otras máquinas de la red para empezar a utilizarlo en aplicaciones mas robustas.

Sin embargo, existen otras reglas que definen quien puede hacer que, las cuales se pueden determinar de acuerdo a la manera en que se manejan cada uno de los archivos de configuración.

Colección de variables, que son accesibles a una comunidad específica.

Tipo de acceso, ya sea de lectura (READ) o de lectura/escritura (WRITE).

Definición del MIB, que impone límites de acceso a secciones particulares.

Otros elementos que te permiten configurar el servidor snmp para un mejor rendimiento.

### Monitorear procesos

Se puede usar un agente para checar algunos procesos que estén corriendo en alguna máquina en específico.

Sintaxis:

```
proc [nombre] [max=0] [min=0]
    nombre      - El nombre del proceso a checar.
    max          - El número máximo de procesos corriendo. Default es 0.
```

min                    - El número mínimo de procesos corriendo. Default es 0.

Por ejemplo, para estar al tanto de que el servidor de montaje (mountd), el servidor de correo (sendmail) esten corriendo, se hace la siguiente instrucción:

```
proc mountd
proc sendmail
```

Para verificar que existan 4 procesos talk como máximo o 1 como mínimo, se hace:

```
proc ntalkd 4 1
```

Cuando el valor mínimo y máximo son ambos 0, se asume que se quiere un maximo infinito y un minimo de 1.

### Monitorear discos

El agente puede checar el espacio libre de los discos habilitados y montados en el sistema.

Sintaxis:

disk [path] [min=espacio\_libre]

path                    - Ruta especifica a un dispositivo de HDD.  
min                    - El espacio libre como mínimo en disco.

Por ejemplo, para checar que en la partición de / existan 10 MB libres por lo menos.

```
disk / 1000
```

### Monitoreo de promedios

Se pueden monitorear las consultas para un tiempo determinado.

Sintaxis:

load [1max] [5max] [15max]

1max                    - Limita a un minuto el tiempo en que se tarda una consulta.  
5max                    - Limita a cinco minuto el tiempo en que se tarda una consulta.  
15max                   - Limita a quince minuto el tiempo en que se tarda una consulta.

Por ejemplo,

```
load 12 14 14
```

### Scripts ejecutables

Se pueden tener programas que ejecute el agente y regresen un valor sencillo en una linea de salida junto con un código de salida.

Sintaxis:

exec [nombre] [programa] [argumentos]

nombre                    - Un nombre genérico  
programa                   - El programa ejecutable para correr, incluyendo la ruta donde se localiza.  
argumentos                   - Argumentos opcionales que se pasen al programa

Ejemplo:

```
exec echotest /bin/echo hola mundo
```

Se puede especificar un shell propio, la limitación se encuentra en que solo se almacena la salida de la primera línea del programa o shell ejecutado.

### Seccion extendible

Este es una extensión a la opción anterior que permite líneas múltiples de salida para un programa o shell, colocando cada línea de salida en una variable del MIB, dentro de su propia tabla.

Se debe modificar el archivo mib.txt para corregir si se desea la descripción de las salidas dentro de un texto razonable en .50.\*.

Sintaxis:

Es la misma a excepción del primer argumento que trata de identificar la rama del árbol en donde se encuentra la tabla del MIB para almacenar las salidas.

```
exec [ID] [nombre] [programa] [argumentos]
```

ID - Identificador del objeto de la tabla del MIB donde se almacenarán las salidas del programa o shell ejecutado.

Ejemplo:

```
exec .1.3.6.1.4.1.2021.50 echotest /bin/echo hola mundo
```

### Paso del control

Pasa el control total de las variables MIB en una porción del MIBOID a un comando para que se ejecute.

Sintaxis:

```
pass miboid comando
```

miboid - Identificador numérico de la rama que se pasará como parámetros al

comando

comando - Comando a ejecutar.

Ejemplo:

```
pass .1.3.6.1.4.1.2021.255 /bin/sh /usr/local/passtest
```

### Control de subagentes

El agente soporta subagentes usando un numero extendible de mecanismos. Por ejemplo, si se desea ocupar el soporte AgentX por default, se debe escribir dentro de smpd.conf como:

```
master agentx
```