

Práctica 1.

Implementación de Sistema de Administración de Red utilizando el Protocolo SNMP

4CM3

Campillo Lopez Jimena Rosalía
Díaz Ortíz Víctor Hugo
Hermenegildo Avendaño Luis Enrique

03 Abril de 2019

Índice

1. Introducción Teórica	3
1.1. Acerca del Protocolo SNMP	4
2. Desarrollo	6
2.1. Instalación de Observium en una máquina virtual	6
2.2. Instalación y configuración de Agentes	8
2.2.1. Instalación y configuración de SNMP en Windows . . .	8
2.2.2. Instalación y configuración de SNMP en Linux	11
2.2.3. Publicación de Agentes	13
3. Cuestionario	17
3.1. Ejercicio MIB	17
3.2. Análisis de tráfico	25
4. Implementación de Modelo	28
4.1. Agregar Agente	28
4.2. Eliminar Agente	28
4.3. Estado del Agente	29
5. Conclusiones	30
6. Referencias Bibliográficas	32

1. Introducción Teórica

Un sistema de gestión de red es una herramienta para monitorear y controlar la red, diseñado para ver la red entera como una arquitectura unificada, con direcciones y etiquetas. Las estaciones de gestión y el agente (equipos) están enlazados por el protocolo de gestión de red, un protocolo SNMP, proyectado para redes basadas en OSI y en TCP/IP.

El SNMP se ha convertido en un estándar de gestión de red dominante y la mayoría de los equipos de interconexión (routers, Switches, Hub, puentes) dispositivos de encaminamiento, estaciones de trabajo y PC ofrecen paquetes de agentes SNMP para ser gestionados.

El SNMP se implementa de una forma fácil y consume un tiempo modesto del procesador y de recursos de red, hoy día se consigue la SNMPV2, no es más que un protocolo que se utiliza para intercambiar información de gestión y definir la estructura de información para servir de apoyo estratégico a la gestión de redes y a la interconexión de ellas.

Las redes tienen una gran importancia ya que mientras más grandes son, tienden a tener sistemas complejos soportando más aplicaciones y usuarios. Conforme van creciendo, se empiezan a descubrir ciertos problemas, como la necesidad de aplicaciones distribuidas para poder compartir recursos y la detección y solución de fallas.

Para dar respuesta a estas necesidades han surgido aplicaciones estándar que permiten administrar las redes, cubriendo servicios, protocolos y bases de información de gestión.

Un sistema de gestión de red es una colección de herramientas para el monitoreo y control de redes el cual está compuesto por:

- Estación de gestión.
- Agente.
- Base de datos de información de gestión.
- Protocolo de gestión de red.

La finalidad de llevar a cabo una administración de redes es dar un servicio para emplear una variedad de herramientas, aplicaciones y dispositivos que sirvan para ayudar en la supervisión y mantenimiento. Cabe destacar que esta tarea recae en un administrador de red, no es más que una persona responsable de supervisar y controlar el hardware y software de la misma,

trabaja en la detección y corrección de problemas que hacen ineficiente o imposible la comunicación.

1.1. Acerca del Protocolo SNMP

SNMP opera en el nivel de aplicación, utilizando el protocolo de transporte TCP/IP, por lo que ignora los aspectos específicos del hardware sobre el que funciona. La gestión se lleva a cabo al nivel de IP, por lo que se pueden controlar dispositivos que estén conectados en cualquier red accesible desde Internet y no únicamente aquellos localizados en la propia red local.

El protocolo SNMP está compuesto por dos elementos: el agente y el gestor. Es una arquitectura cliente-servidor, en la cual el agente desempeña el papel de servidor y el gestor el de cliente.

El **agente** es un programa que se ejecuta en cada nodo de red que se desea gestionar o monitorizar. Ofrece una interfaz de todos los elementos que se pueden configurar, estos elementos se almacenan en unas estructuras de datos llamadas "Management Information Base" (MIB). El cual representa la parte del servidor, en la medida que tiene la información que se desea gestionar y espera comandos por parte del cliente.

El **gestor** es el software que se ejecuta en la estación encargada de monitorizar la red, y su tarea consiste en consultar los diferentes agentes que se encuentran en los nodos de la red los datos que estos han ido obteniendo.

En esencia, el SNMP es un protocolo muy sencillo puesto que todas las operaciones se realizan bajo el paradigma de carga y almacenamiento (load-and-store), lo que permite un juego de comandos reducido. Un gestor puede realizar sólo dos tipos diferentes de operaciones sobre un agente: leer o escribir un valor de una variable en el MIB del agente. Estas dos operaciones se conocen como petición de lectura (get-request) y petición de escritura (set-request). Hay un comando para responder a una petición de lectura llamado respuesta de lectura (get-response), que es utilizado únicamente por el agente.

La posibilidad de ampliación del protocolo está directamente relacionado con la capacidad del MIB de almacenar nuevos elementos, si un fabricante quiere añadir un nuevo comando a un dispositivo, como puede ser un enrutador, tan sólo tiene que añadir las variables correspondientes a su base de datos (MIB).

Casi todos los fabricantes implementan versiones agente de SNMP en sus dispositivos: enrutadores, hubs, sistemas operativos, etc. Linux no es una

excepción, existen varios agentes SNMP disponibles públicamente.

2. Desarrollo

En esta sección explicaremos como instalar y configurar el protocolo SNMP tanto el Windows como Linux, además, explicaremos como agregar los agentes al gestor Observium.

2.1. Instalación de Observium en una máquina virtual

Abrimos VirtualBox y seleccionamos en el menú la opción que dice "Servicio Virtualizado". Seleccionamos la ruta donde tenemos guardada la máquina virtual.

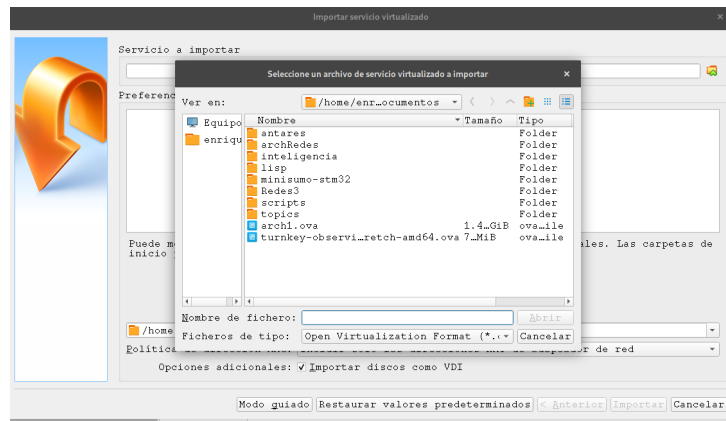


Figura 1: Instalación Observium

Seguimos las instrucciones de instalación, y así se configurará Observium con una IP como se muestra en la captura de pantalla.

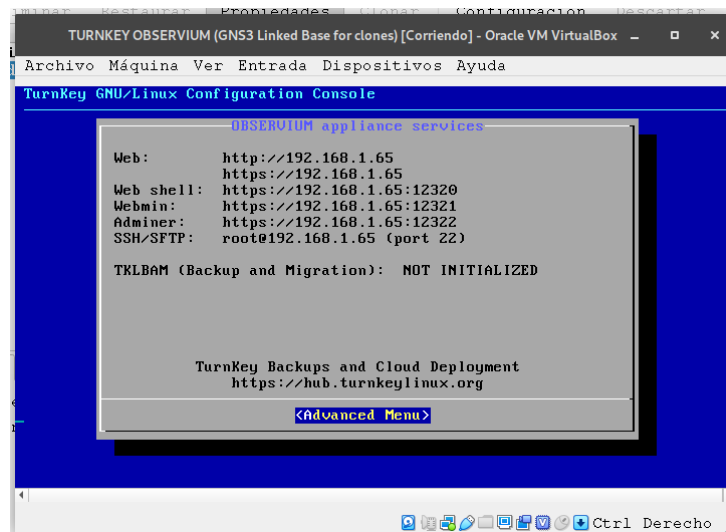


Figura 2: Instalación Observium

A continuación abrimos nuestro Explorador Web e introducimos la IP correspondiente al equipo, que en nuestro caso es **192.168.1.65**. Procedemos a introducir nuestro usuario y contraseña.

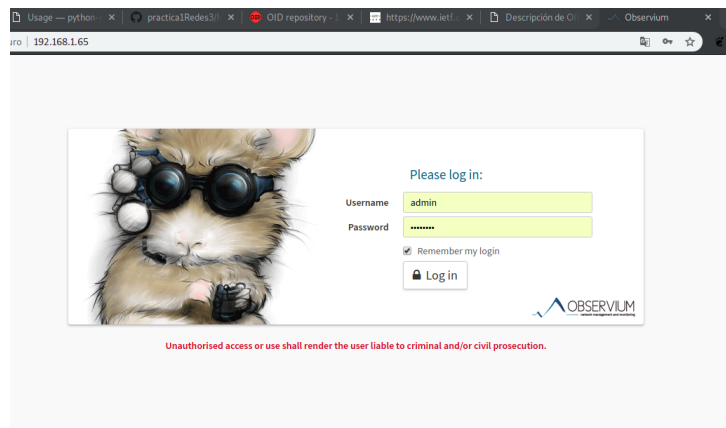


Figura 3: Ingreso

De esta forma queda configurado nuestro Observium.

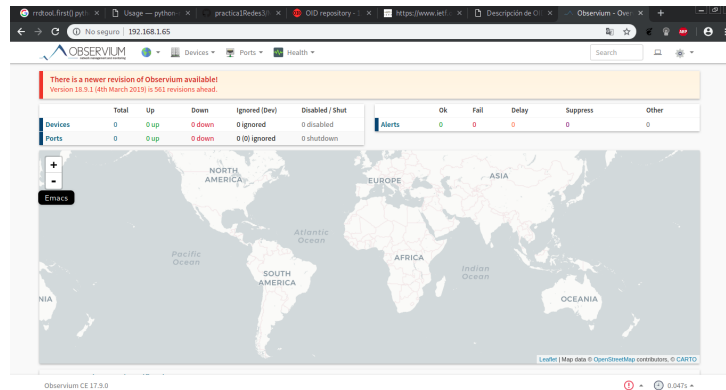


Figura 4: Instalación finalizada

2.2. Instalación y configuración de Agentes

2.2.1. Instalación y configuración de SNMP en Windows

Iniciamos sesión en nuestro Sistema Operativo Windows, en nuestro caso utilizamos la versión XP Service Pack 3, de 32 bits. Para instalar nuestro agente nos dirigimos al menú Panel de Control -¿agregar o remover programas -¿agregar o remover componentes de Windows y seleccionamos la opción de servicios de red -¿continuar. Le damos en la opción siguiente y de esta forma se instalarán los paquetes necesarios para el agente SNMP.

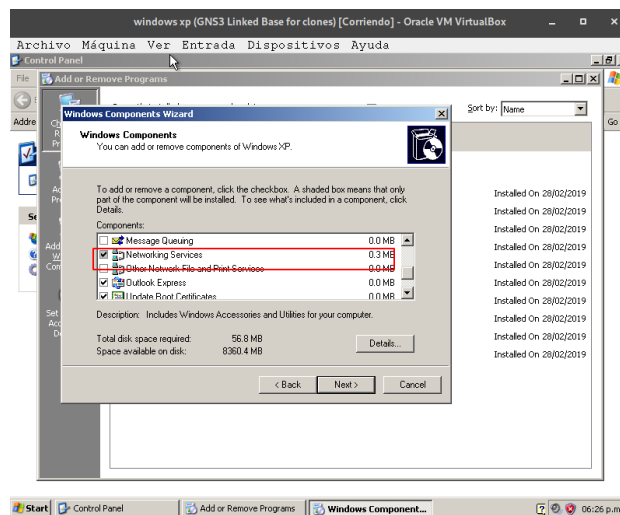


Figura 5: SNMP en Windows

Para configurar nuestro agente, en el menú Inicio seleccionamos la opción de "Computer Management", una vez ahí seleccionamos la opción de "Servicios y aplicaciones", del lado derecho seleccionamos "SNMP Service". Al darle click nos abrirá una nueva ventana donde procederemos a introducir nuestra configuración.

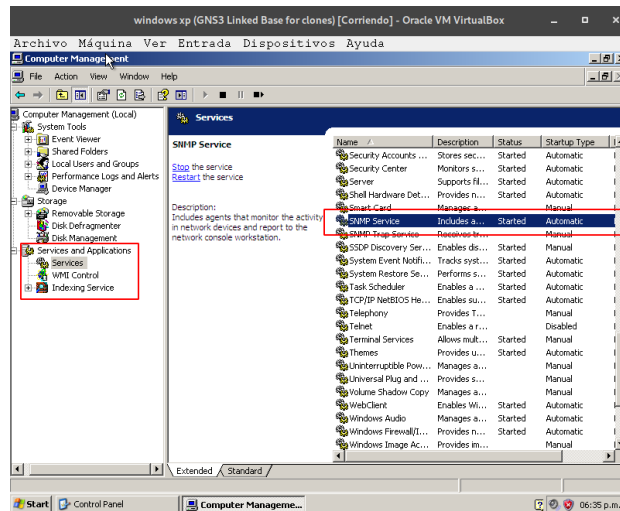


Figura 6: SNMP en Windows

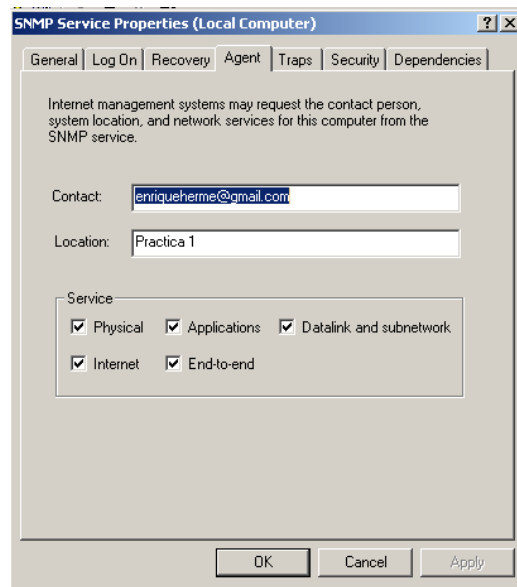


Figura 7: SNMP en Windows

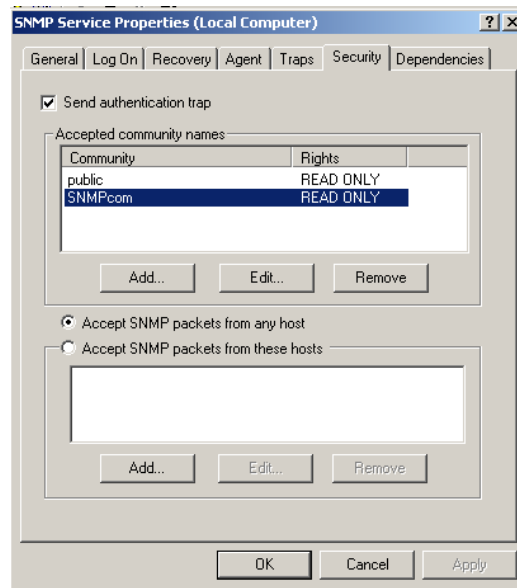


Figura 8: SNMP en Windows

Podemos observar, que nosotros configuramos nuestra comunidad al nombre de "SNMPcom".

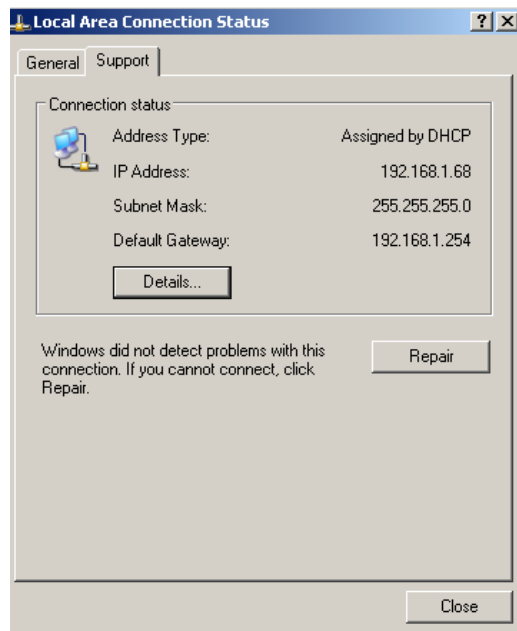


Figura 9: SNMP en Windows

Nuestra instalación y configuración están finalizadas.

2.2.2. Instalación y configuración de SNMP en Linux

Una vez logeados en nuestra sesión, procedemos a instalar el servicio SNMP con el comando `pacman -S net-snmp`. En este caso el paquete ya se encuentra instalado, entonces checamos el estado del servicio con el comando `systemctl status snmpd`.

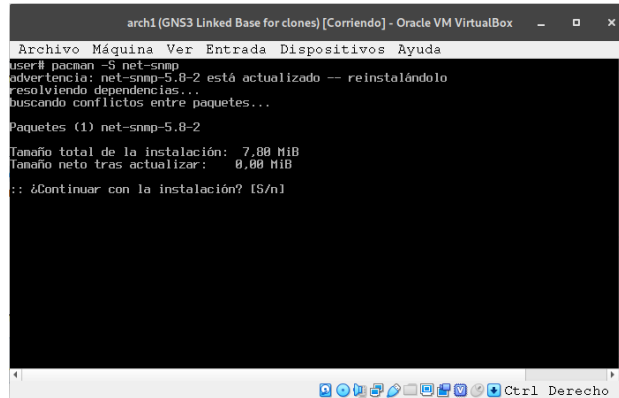


Figura 10: SNMP en Linux

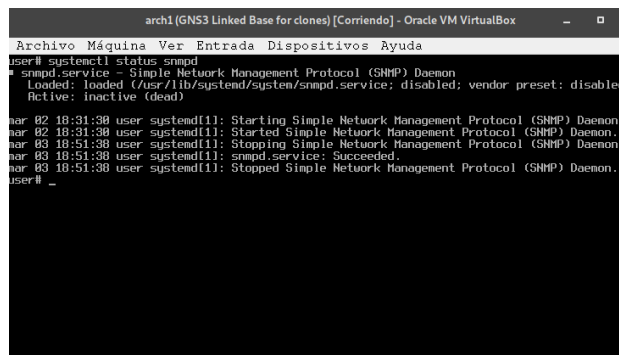


Figura 11: SNMP en Windows

Procedemos a configurar el servicio, para crear el servicio creamos un directorio con la siguiente ruta:

```
mkdir /etc/snmp/
```

Introducimos el siguiente comando para crear nuestra comunidad, en nuestro caso `SNMPcom`.

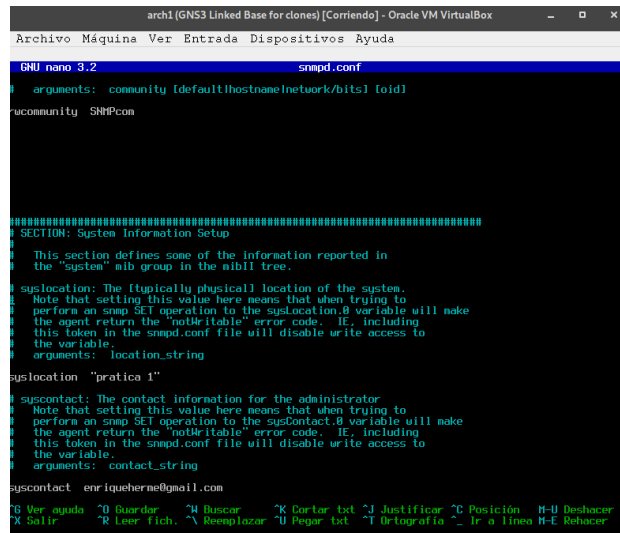
```
echo rocommunity read_only_community_string >> /etc/snmp/snmpd.conf
```

echo rocommunity SNMPcom /etc/snmp/snmpd.conf

Creamos un respaldo del archivo original.

mv /etc/snmp/snmpd.conf /etc/snmp/snmpd.conf.bck

Ahora procedemos a configurar el archivo `snmpd.conf`



```

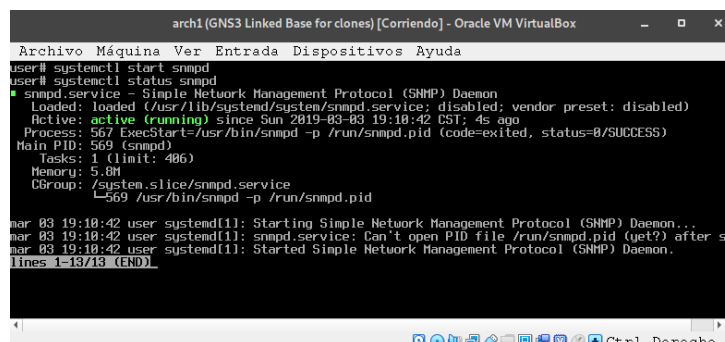
arch1 (GN53 Linked Base for clones) [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
GNU nano 3.2 snmpd.conf
# arguments: community [default:hostname.network.bits] [oid]
rocommunity SNMPcom

=====
# SECTION: System Information Setup
#
# This section defines some of the information reported in
# the "system" mib group in the mibII tree.
#
# syslocation: The (typically physical) location of the system.
# Note that setting this value here means that when trying to
# perform an snmp SET operation to the syslocation.0 variable will make
# the agent return the "notWritable" error code. If, including
# this token in the snmpd.conf file will disable write access to
# the variable.
# arguments: location_string
syslocation "pratica 1"
#
# syscontact: The contact information for the administrator
# Note that setting this value here means that when trying to
# perform an snmp SET operation to the syscontact.0 variable will make
# the agent return the "notWritable" error code. If, including
# this token in the snmpd.conf file will disable write access to
# the variable.
# arguments: contact_string
syscontact enriqueherme@gmail.com
#
# Ver ayuda  ?D Guardar  ?W Buscar  ?K Cortar txt  ?J Justificar  ?C Posición  M-U Deshacer
# Salir  ?R Leer Fich.  \ Reemplazar  ?U Pegar txt  ?T Ortografía  _ Ir a línea M-E Rehacer

```

Figura 12: SNMP en Linux

Una vez modificado el archivo, la configuración básica queda de la siguiente manera: **rwcommunity SNMPcom**, **syslocation “practical”**, **syscontact enriqueherme@gmail.com** Procedemos a iniciar el servicio con el comando **systemctl start snmpd** y para comprobar que efectivamente el servicio está activo, tecleamos **systemctl status snmpd**



```

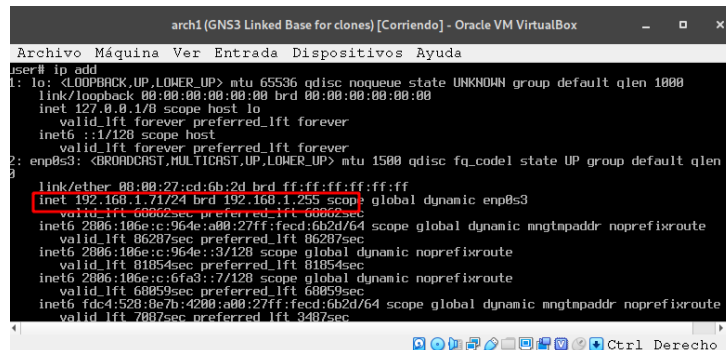
arch1 (GN53 Linked Base for clones) [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
user# systemctl start snmpd
user# systemctl status snmpd
● snmpd.service - Simple Network Management Protocol (SNMP) Daemon
   Loaded: loaded (/usr/lib/systemd/system/snmpd.service; disabled; vendor preset: disabled)
   Active: active (running) since Sun 2019-03-03 19:10:42 CST; 4s ago
     Process: 567 ExecStart=/usr/bin/snmpd -p /run/snmpd.pid (code=exited, status=0/SUCCESS)
    Main PID: 569 (snmpd)
       Tasks: 1 (limit: 486)
      Memory: 5.8M
     CGroup: /system.slice/snmpd.service
            └─569 /usr/bin/snmpd -p /run/snmpd.pid

mar 03 19:10:42 user systemd[1]: Starting Simple Network Management Protocol (SNMP) Daemon...
mar 03 19:10:42 user systemd[1]: snmpd.service: Can't open PID file /run/snmpd.pid (yet?) after s
mar 03 19:10:42 user systemd[1]: Started Simple Network Management Protocol (SNMP) Daemon.
lines 1-13/13 (END)

```

Figura 13: SNMP en Linux

Por último, checamos la IP del dispositivo. Nuestra instalación y configuración están terminadas.



```

arch1(GNS3 Linked Base for clones) [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
user# ip add
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp8s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:cd:6b:2d brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.255/24 brd 192.168.1.255 scope global dynamic enp8s3
        valid_lft 60000sec preferred_lft 60000sec
    inet6 2006:106e:c:964e:a00:27ff:fed:6b2d/64 scope global dynamic mngtnpaddr noprefixroute
        valid_lft 86287sec preferred_lft 86287sec
    inet6 2006:106e:c:964e::3/128 scope global dynamic noprefixroute
        valid_lft 81854sec preferred_lft 81854sec
    inet6 2006:106e:c:6fa9::7/128 scope global dynamic noprefixroute
        valid_lft 60059sec preferred_lft 60059sec
    inet6 fdc4:528:8e7b:4200:a00:27ff:fed:6b2d/64 scope global dynamic mngtnpaddr noprefixroute
        valid_lft 70077sec preferred_lft 3487sec
    
```

Figura 14: SNMP en Linux

2.2.3. Publicación de Agentes

Para agregar un agente a nuestro gestor, nos dirigimos a Observium, damos clic en el menú Devices -> Add devices. Agregamos la IP del agente en turno a agregar.

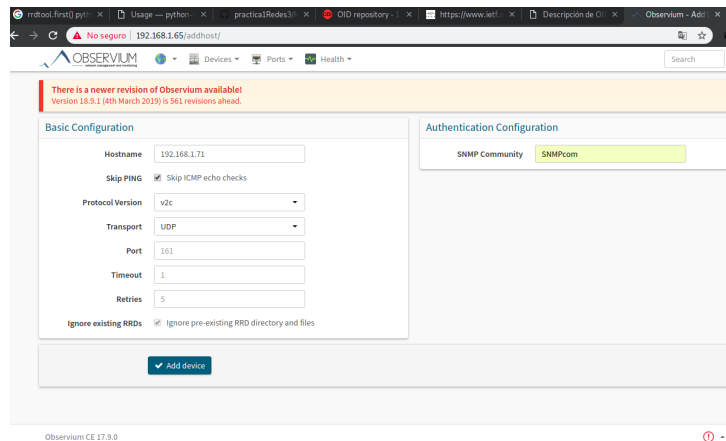


Figura 15: Agente Linux en Observium

Una vez agregado con éxito, nos aparecerá esta ventana.

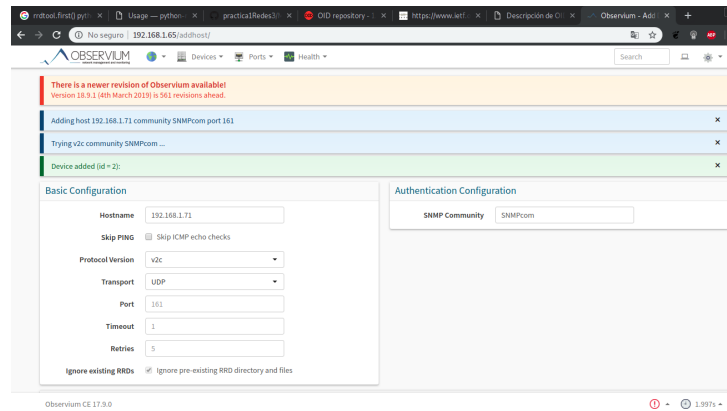


Figura 16: Agente Linux en Observium

Ahora, agregamos el agente Windows con la IP 192.168.1.68.

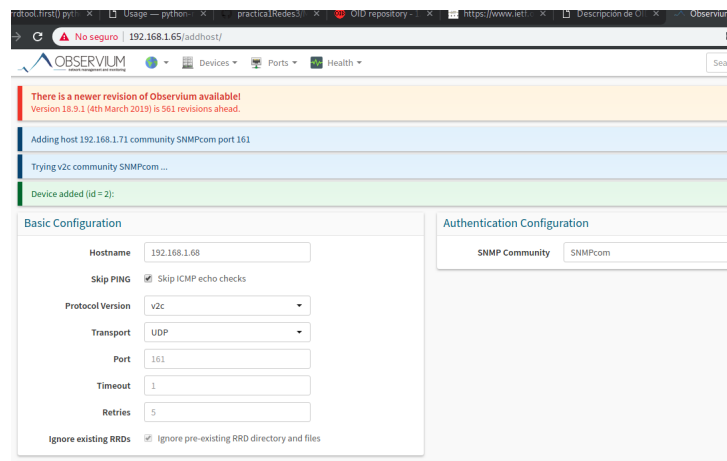


Figura 17: Agente Windows en Observium

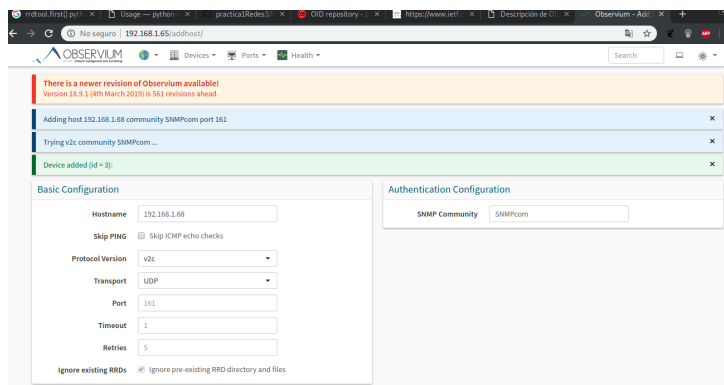


Figura 18: Agente Windows en Observium

Observamos los agentes que tenemos ya registrados

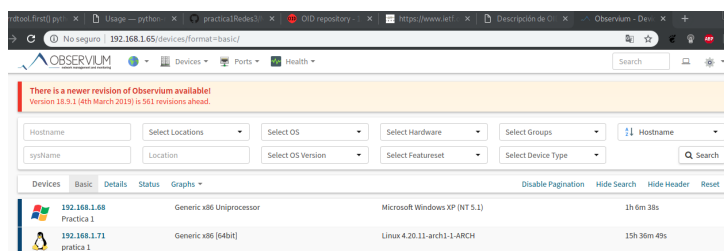


Figura 19: Agentes en Observium

Agente Windows publicado Aquí podemos observar toda la información que Observium obtiene sobre el agente.

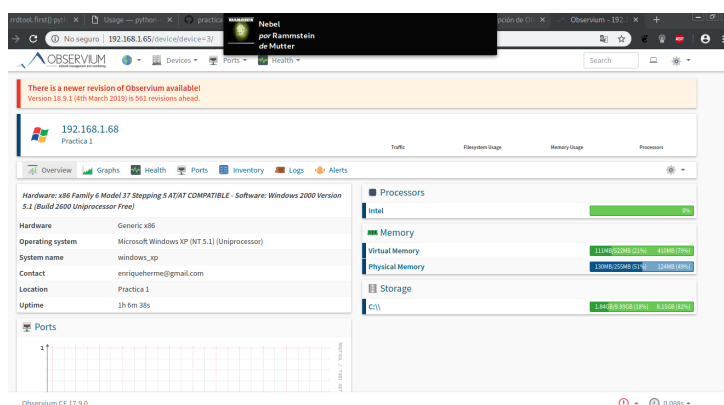


Figura 20: Agente Windows

Agente Linux publicado Aquí podemos observar toda la información que Observium obtiene sobre el agente.

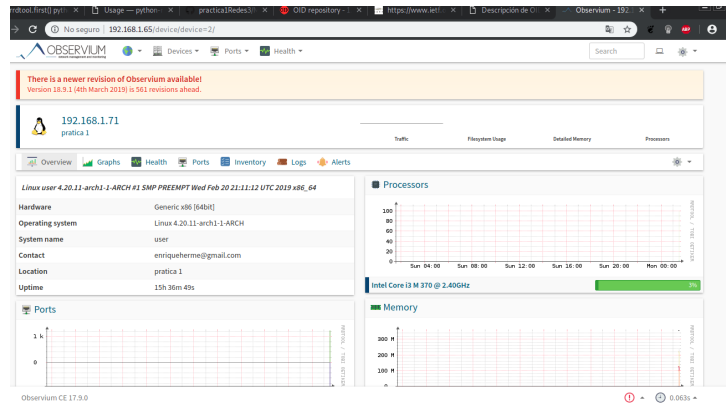


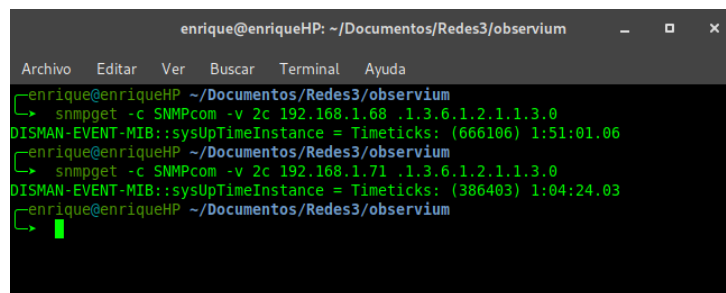
Figura 21: Agente Linux

3. Cuestionario

En la siguiente sección resolveremos un cuestionario haciendo las consultas correspondientes a los objetos, desde nuestro agente Windows y Linux.

3.1. Ejercicio MIB

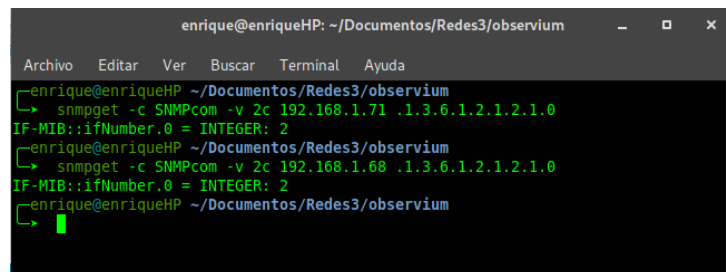
1. ¿Cuándo fue el último reinicio (Día, hora y minuto) de los agentes?



```
enrique@enriqueHP: ~/Documentos/Redes3/observium
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
enrique@enriqueHP ~/Documentos/Redes3/observium
→ snmpget -c SNMPcom -v 2c 192.168.1.68 .1.3.6.1.2.1.1.3.0
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (666106) 1:51:01.06
enrique@enriqueHP ~/Documentos/Redes3/observium
→ snmpget -c SNMPcom -v 2c 192.168.1.71 .1.3.6.1.2.1.1.3.0
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (386403) 1:04:24.03
enrique@enriqueHP ~/Documentos/Redes3/observium
```

Figura 22: Ejercicio MIB

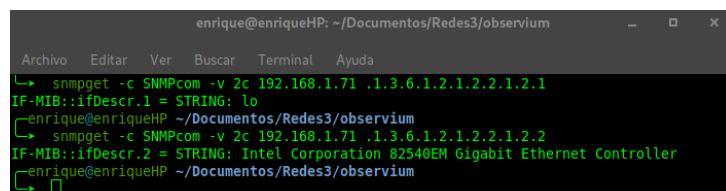
2. ¿Cuántas interfaces Ethernet tienen?



```
enrique@enriqueHP: ~/Documentos/Redes3/observium
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
enrique@enriqueHP ~/Documentos/Redes3/observium
→ snmpget -c SNMPcom -v 2c 192.168.1.71 .1.3.6.1.2.1.2.1.0
IF-MIB::ifNumber.0 = INTEGER: 2
enrique@enriqueHP ~/Documentos/Redes3/observium
→ snmpget -c SNMPcom -v 2c 192.168.1.68 .1.3.6.1.2.1.2.1.0
IF-MIB::ifNumber.0 = INTEGER: 2
enrique@enriqueHP ~/Documentos/Redes3/observium
```

Figura 23: Ejercicio MIB

- a) Para conocer las interfaces, hicimos una consulta más. **Windows**



```
enrique@enriqueHP: ~/Documentos/Redes3/observium
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
enrique@enriqueHP ~/Documentos/Redes3/observium
→ snmpget -c SNMPcom -v 2c 192.168.1.71 .1.3.6.1.2.1.2.2.1.2.1
IF-MIB::ifDescr.1 = STRING: lo
enrique@enriqueHP ~/Documentos/Redes3/observium
→ snmpget -c SNMPcom -v 2c 192.168.1.71 .1.3.6.1.2.1.2.2.1.2.2
IF-MIB::ifDescr.2 = STRING: Intel Corporation 82540EM Gigabit Ethernet Controller
enrique@enriqueHP ~/Documentos/Redes3/observium
```

Figura 24: Ejercicio MIB

b) Para conocer las interfaces, hicimos una consulta más. **Linux**

```

enrique@enriqueHP: ~/Documentos/Redes3/observium
Archivo Editar Ver Buscar Terminal Ayuda
enrique@enriqueHP ~/Documentos/Redes3/observium
→ snmpget -c SNMPcom -v 2c 192.168.1.68 .1.3.6.1.2.1.2.2.1.5.1
IF-MIB::ifDescr.1 = STRING: MS TCP Loopback interface.
enrique@enriqueHP ~/Documentos/Redes3/observium

```

Figura 25: Ejercicio MIB

3. ¿Cuál es la velocidad (en MBPS) de esas interfaces?

a) **Linux**

```

enrique@enriqueHP: ~/Documentos/Redes3/observium
Archivo Editar Ver Buscar Terminal Ayuda
enrique@enriqueHP ~/Documentos/Redes3/observium
→ snmpget -c SNMPcom -v 2c 192.168.1.71 .1.3.6.1.2.1.2.2.1.5.1
IF-MIB::ifSpeed.1 = Gauge32: 100000000
enrique@enriqueHP ~/Documentos/Redes3/observium
→ snmpget -c SNMPcom -v 2c 192.168.1.71 .1.3.6.1.2.1.2.2.1.5.2
IF-MIB::ifSpeed.2 = Gauge32: 1000000000
enrique@enriqueHP ~/Documentos/Redes3/observium

```

Figura 26: Ejercicio MIB

b) **Windows**

```

enrique@enriqueHP: ~/Documentos/Redes3/observium
Archivo Editar Ver Buscar Terminal Ayuda
enrique@enriqueHP ~/Documentos/Redes3/observium
→ snmpget -c SNMPcom -v 2c 192.168.1.68 .1.3.6.1.2.1.2.2.1.5.2
Error in packet
Reason: (noSuchName) There is no such variable name in this MIB.
Failed object: IF-MIB::ifSpeed.2

```

Figura 27: Ejercicio MIB

4. ¿Cuál es la interfaz que ha recibido el mayor número de octetos?

a) En Linux tenemos dos interfaces. Como vemos, la interfaz no.2 es la que tiene el mayor número de octetos.

```

enrique@enriqueHP: ~/Documentos/Redes3/observium
Archivo Editar Ver Buscar Terminal Ayuda
enrique@enriqueHP ~/Documentos/Redes3/observium
→ snmpget -c SNMPcom -v 2c 192.168.1.71 .1.3.6.1.2.1.2.2.1.10.1
IF-MIB::ifInOctets.1 = Counter32: 0
enrique@enriqueHP ~/Documentos/Redes3/observium
→ snmpget -c SNMPcom -v 2c 192.168.1.71 .1.3.6.1.2.1.2.2.1.10.2
IF-MIB::ifInOctets.2 = Counter32: 15925475
enrique@enriqueHP ~/Documentos/Redes3/observium

```

Figura 28: Ejercicio MIB

- b) En Windows solo tenemos una interfaz, por lo que será automáticamente la que mayor número de octetos recibe.

```

enrique@enriqueHP: ~/Documentos/Redes3/observium
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
enrique@enriqueHP ~/Documentos/Redes3/observium
> snmpget -c SNMPcom -v 2c 192.168.1.68 .1.3.6.1.2.1.2.2.1.10.1
IF-MIB::ifInOctets.1 = Counter32: 2376
enrique@enriqueHP ~/Documentos/Redes3/observium

```

Figura 29: Ejercicio MIB

5. ¿Cuál es la MAC de esa interfaz?

- a) **Linux**

```

enrique@enriqueHP: ~/Documentos/Redes3/observium
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
enrique@enriqueHP ~/Documentos/Redes3/observium
> snmpget -c SNMPcom -v 2c 192.168.1.71 .1.3.6.1.2.1.2.2.1.6.2
IF-MIB::ifPhysAddress.2 = STRING: 8:0:27:cd:6b:2d
enrique@enriqueHP ~/Documentos/Redes3/observium

```

Figura 30: Ejercicio MIB

- 1) Comprobamos información mediante el uso de herramienta externa.

```

arch1 (GNS3 Linked Base for clones) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
user@ip add
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:cd:6b:2d brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.71/24 brd 192.168.1.255 scope global dynamic enp0s3
        valid_lft 60950sec preferred_lft 60950sec
    inet6 2006:106e:c:964e:a00:27ff:fe0d:6b2d/64 scope global dynamic mngtaddr noprefixroute
        valid_lft 86399sec preferred_lft 86399sec
    inet6 2006:106e:c:964e::3/128 scope global dynamic noprefixroute
        valid_lft 74743sec preferred_lft 74743sec
    inet6 2006:106e:c:6fa3::7/128 scope global dynamic noprefixroute
        valid_lft 60947sec preferred_lft 60947sec
    inet6 fd4:528:8e7b:4280:a00:27ff:fe0d:6b2d/64 scope global dynamic mngtaddr noprefixroute
        valid_lft 7199sec preferred_lft 3599sec
    inet6 fe80::a00:27ff:fe0d:6b2d/64 scope link

```

Figura 31: Ejercicio MIB

- b) **Windows**

```
enrique@enriqueHP: ~/Documentos/Redes3/observium
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
└─ snmpget -c SNMPcom -v 2c 192.168.1.68 .1.3.6.1.2.1.2.2.1.6.1
IF-MIB::ifPhysAddress.1 = STRING:
enrique@enriqueHP ~/Documentos/Redes3/observium
└─ snmpget -c SNMPcom -v 2c 192.168.1.68 .1.3.6.1.2.1.2.2.1.6.2
Error in packet
Reason: (noSuchName) There is no such variable name in this MIB.
Failed object: IF-MIB::ifPhysAddress.2
```

Figura 32: Ejercicio MIB

- 1) Comprobamos información mediante el uso de herramienta externa.

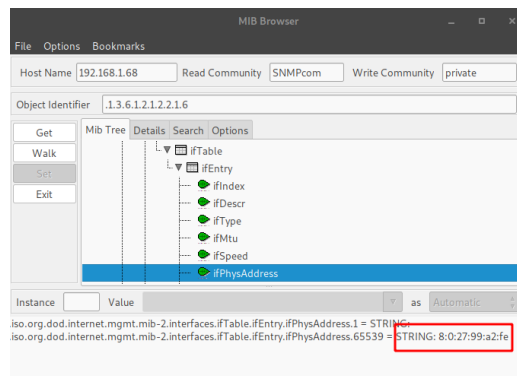


Figura 33: Ejercicio MIB

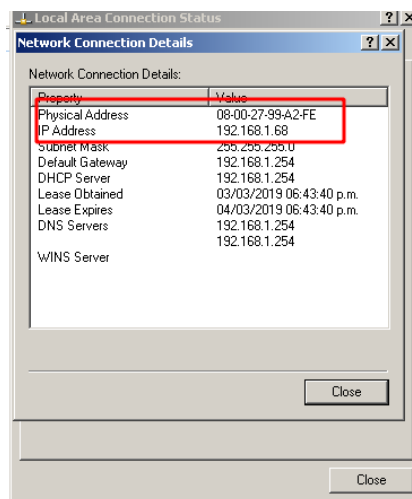
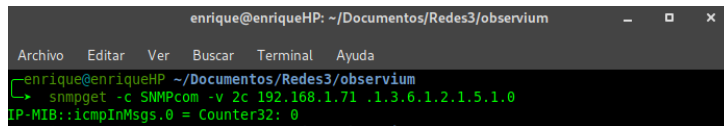


Figura 34: Ejercicio MIB

6. ¿Cuántos mensajes ICMP ha recibido el agente?

a) **Linux**



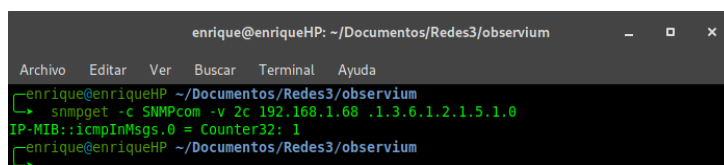
```

enrique@enriqueHP: ~/Documentos/Redes3/observium
Archivo Editar Ver Buscar Terminal Ayuda
enrique@enriqueHP ~/Documentos/Redes3/observium
➤ snmpget -c SNMPcom -v 2c 192.168.1.71 .1.3.6.1.2.1.5.1.0
IP-MIB::icmpInMsgs.0 = Counter32: 0

```

Figura 35: Ejercicio MIB

b) **Windows**



```

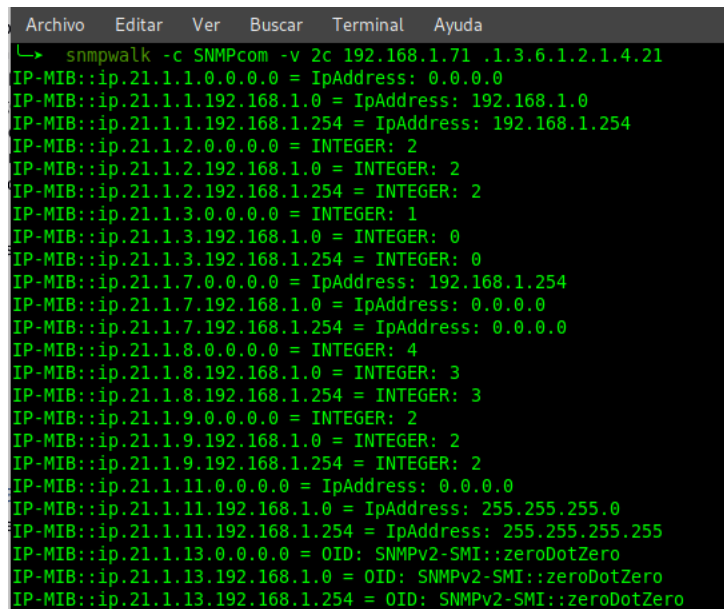
enrique@enriqueHP: ~/Documentos/Redes3/observium
Archivo Editar Ver Buscar Terminal Ayuda
enrique@enriqueHP ~/Documentos/Redes3/observium
➤ snmpget -c SNMPcom -v 2c 192.168.1.68 .1.3.6.1.2.1.5.1.0
IP-MIB::icmpInMsgs.0 = Counter32: 1
enrique@enriqueHP ~/Documentos/Redes3/observium

```

Figura 36: Ejercicio MIB

7. ¿Cuántas entradas tiene la tabla de enrutamiento IP?

a) **Linux**



```

Archivo Editar Ver Buscar Terminal Ayuda
➤ snmpwalk -c SNMPcom -v 2c 192.168.1.71 .1.3.6.1.2.1.4.21
IP-MIB::ip.21.1.1.0.0.0.0 = IPAddress: 0.0.0.0
IP-MIB::ip.21.1.1.192.168.1.0 = IPAddress: 192.168.1.0
IP-MIB::ip.21.1.1.192.168.1.254 = IPAddress: 192.168.1.254
IP-MIB::ip.21.1.2.0.0.0.0 = INTEGER: 2
IP-MIB::ip.21.1.2.192.168.1.0 = INTEGER: 2
IP-MIB::ip.21.1.2.192.168.1.254 = INTEGER: 2
IP-MIB::ip.21.1.3.0.0.0.0 = INTEGER: 1
IP-MIB::ip.21.1.3.192.168.1.0 = INTEGER: 0
IP-MIB::ip.21.1.3.192.168.1.254 = INTEGER: 0
IP-MIB::ip.21.1.7.0.0.0.0 = IPAddress: 192.168.1.254
IP-MIB::ip.21.1.7.192.168.1.0 = IPAddress: 0.0.0.0
IP-MIB::ip.21.1.7.192.168.1.254 = IPAddress: 0.0.0.0
IP-MIB::ip.21.1.8.0.0.0.0 = INTEGER: 4
IP-MIB::ip.21.1.8.192.168.1.0 = INTEGER: 3
IP-MIB::ip.21.1.8.192.168.1.254 = INTEGER: 3
IP-MIB::ip.21.1.9.0.0.0.0 = INTEGER: 2
IP-MIB::ip.21.1.9.192.168.1.0 = INTEGER: 2
IP-MIB::ip.21.1.9.192.168.1.254 = INTEGER: 2
IP-MIB::ip.21.1.11.0.0.0.0 = IPAddress: 0.0.0.0
IP-MIB::ip.21.1.11.192.168.1.0 = IPAddress: 255.255.255.0
IP-MIB::ip.21.1.11.192.168.1.254 = IPAddress: 255.255.255.255
IP-MIB::ip.21.1.13.0.0.0.0 = OID: SNMPv2-SMI::zeroDotZero
IP-MIB::ip.21.1.13.192.168.1.0 = OID: SNMPv2-SMI::zeroDotZero
IP-MIB::ip.21.1.13.192.168.1.254 = OID: SNMPv2-SMI::zeroDotZero

```

Figura 37: Ejercicio MIB

b) **Windows**

```

Archivo  Editar  Ver  Buscar  Terminal  Ayuda
enrique@enriqueHP ~/Documentos/Redes3/observium
└─ snmpwalk -c SNMPcom -v 2c 192.168.1.71 .1.3.6.1.2.1.4.21
enrique@enriqueHP ~/Documentos/Redes3/observium
└─ snmpwalk -c SNMPcom -v 2c 192.168.1.68 .1.3.6.1.2.1.4.21
IP-MIB::ip.21.1.1.0.0.0.0 = IPAddress: 0.0.0.0
IP-MIB::ip.21.1.1.127.0.0.0 = IPAddress: 127.0.0.0
IP-MIB::ip.21.1.1.192.168.1.0 = IPAddress: 192.168.1.0
IP-MIB::ip.21.1.1.192.168.1.68 = IPAddress: 192.168.1.68
IP-MIB::ip.21.1.1.192.168.1.255 = IPAddress: 192.168.1.255
IP-MIB::ip.21.1.1.224.0.0.0 = IPAddress: 224.0.0.0
IP-MIB::ip.21.1.1.255.255.255.255 = IPAddress: 255.255.255.255
IP-MIB::ip.21.1.2.0.0.0.0 = INTEGER: 65539
IP-MIB::ip.21.1.2.127.0.0.0 = INTEGER: 1
IP-MIB::ip.21.1.2.192.168.1.0 = INTEGER: 65539
IP-MIB::ip.21.1.2.192.168.1.68 = INTEGER: 1
IP-MIB::ip.21.1.2.192.168.1.255 = INTEGER: 65539
IP-MIB::ip.21.1.2.224.0.0.0 = INTEGER: 65539
IP-MIB::ip.21.1.2.255.255.255.255 = INTEGER: 65539
IP-MIB::ip.21.1.3.0.0.0.0 = INTEGER: 10
IP-MIB::ip.21.1.3.127.0.0.0 = INTEGER: 1
IP-MIB::ip.21.1.3.192.168.1.0 = INTEGER: 10
IP-MIB::ip.21.1.3.192.168.1.68 = INTEGER: 10
IP-MIB::ip.21.1.3.192.168.1.255 = INTEGER: 10
IP-MIB::ip.21.1.3.224.0.0.0 = INTEGER: 10
IP-MIB::ip.21.1.3.255.255.255.255 = INTEGER: 1
IP-MIB::ip.21.1.4.0.0.0.0 = INTEGER: -1
IP-MIB::ip.21.1.4.127.0.0.0 = INTEGER: -1
IP-MIB::ip.21.1.4.192.168.1.0 = INTEGER: -1
IP-MIB::ip.21.1.4.192.168.1.68 = INTEGER: -1
IP-MIB::ip.21.1.4.192.168.1.255 = INTEGER: -1
IP-MIB::ip.21.1.4.224.0.0.0 = INTEGER: -1
IP-MIB::ip.21.1.4.255.255.255.255 = INTEGER: -1
IP-MIB::ip.21.1.5.0.0.0.0 = INTEGER: -1
IP-MIB::ip.21.1.5.127.0.0.0 = INTEGER: -1
IP-MIB::ip.21.1.5.192.168.1.0 = INTEGER: -1
IP-MIB::ip.21.1.5.192.168.1.68 = INTEGER: -1
IP-MIB::ip.21.1.5.192.168.1.255 = INTEGER: -1
IP-MIB::ip.21.1.5.224.0.0.0 = INTEGER: -1
IP-MIB::ip.21.1.5.255.255.255.255 = INTEGER: -1
IP-MIB::ip.21.1.6.0.0.0.0 = INTEGER: -1
IP-MIB::ip.21.1.6.127.0.0.0 = INTEGER: -1
IP-MIB::ip.21.1.6.192.168.1.0 = INTEGER: -1
IP-MIB::ip.21.1.6.192.168.1.68 = INTEGER: -1
IP-MIB::ip.21.1.6.192.168.1.255 = INTEGER: -1
IP-MIB::ip.21.1.6.224.0.0.0 = INTEGER: -1
IP-MIB::ip.21.1.6.255.255.255.255 = INTEGER: -1
IP-MIB::ip.21.1.7.0.0.0.0 = IPAddress: 192.168.1.254
IP-MIB::ip.21.1.7.127.0.0.0 = IPAddress: 127.0.0.1
IP-MIB::ip.21.1.7.192.168.1.0 = IPAddress: 192.168.1.68
IP-MIB::ip.21.1.7.192.168.1.68 = IPAddress: 127.0.0.1
IP-MIB::ip.21.1.7.192.168.1.255 = IPAddress: 192.168.1.68
IP-MIB::ip.21.1.7.224.0.0.0 = IPAddress: 192.168.1.68
IP-MIB::ip.21.1.7.255.255.255.255 = IPAddress: 192.168.1.68
IP-MIB::ip.21.1.8.0.0.0.0 = INTEGER: 4
IP-MIB::ip.21.1.8.127.0.0.0 = INTEGER: 3
IP-MIB::ip.21.1.8.192.168.1.0 = INTEGER: 3
IP-MIB::ip.21.1.8.192.168.1.68 = INTEGER: 3
IP-MIB::ip.21.1.8.192.168.1.255 = INTEGER: 3
IP-MIB::ip.21.1.8.224.0.0.0 = INTEGER: 3

```

Figura 38: Ejercicio MIB

8. ¿Cuántos datagramas UDP ha recibido el agente?

a) **Linux**

```

Archivo  Editar  Ver  Buscar  Terminal  Ayuda
enrique@enriqueHP ~/Documentos/Redes3/observium
└─▶ snmpget -c SNMPcom -v 2c 192.168.1.71 .1.3.6.1.2.1.7.1.0
UDP-MIB::udpInDatagrams.0 = Counter32: 8823

```

Figura 39: Ejercicio MIB

b) Windows

```

Archivo  Editar  Ver  Buscar  Terminal  Ayuda
enrique@enriqueHP ~/Documentos/Redes3/observium
└─▶ snmpget -c SNMPcom -v 2c 192.168.1.68 .1.3.6.1.2.1.7.1.0
UDP-MIB::udpInDatagrams.0 = Counter32: 2613
enrique@enriqueHP ~/Documentos/Redes3/observium
└─▶

```

Figura 40: Ejercicio MIB

9. ¿Cuántos mensajes EGP ha recibido el agente?

```

Archivo  Editar  Ver  Buscar  Terminal  Ayuda
enrique@enriqueHP ~/Documentos/Redes3/observium
└─▶ snmpget -c SNMPcom -v 2c 192.168.1.71 .1.3.6.1.2.1.6.10.0
TCP-MIB::tcpInSegs.0 = Counter32: 16
enrique@enriqueHP ~/Documentos/Redes3/observium
└─▶ snmpget -c SNMPcom -v 2c 192.168.1.68 .1.3.6.1.2.1.6.10.0
TCP-MIB::tcpInSegs.0 = Counter32: 166

```

Figura 41: Ejercicio MIB

10. Indica el Sistema Operativo que del agente.

```

Archivo  Editar  Ver  Buscar  Terminal  Ayuda
SNMPv2-MIB::sysDescr.0 = STRING: Linux user 4.20.11-arch1-1-ARCH #1 SMP PREEMPT Wed Feb 20 21:11:12 UTC 2019 x86_64
enrique@enriqueHP ~/Documentos/Redes3/observium
└─▶ snmpget -c SNMPcom -v 2c 192.168.1.68 .1.3.6.1.2.1.1.0
SNMPv2-MIB::sysDescr.0 = STRING: Hardware: x86 Family 6 Model 37 Stepping 5 AT/AT COMPATIBLE - Software: Windows 2000 Version 5.1 (Build 2600 Uniprocessor Free)

```

Figura 42: Ejercicio MIB

11. Modifica el nombre del contacto o la ubicación del sistema de un agente.

```

Archivo  Editar  Ver  Buscar  Terminal  Ayuda
enrique@enriqueHP ~/Documentos/Redes3/observium
└─▶ snmpset -c SNMPcom -v 2c 192.168.1.68 .1.3.6.1.2.1.1.4.0
SNMPv2-MIB::sysContact.0 = STRING: enriqueherme@gmail.com
enrique@enriqueHP ~/Documentos/Redes3/observium
└─▶

```

Figura 43: Ejercicio MIB

a) Cambiaremos el contacto por la cadena: **equipo12@gmail.com**

```

Archivo  Editar  Ver  Buscar  Terminal  Ayuda
enrique@enriqueHP ~/Documentos/Redes3/observium
└─ snmpset -v 2c -c SNMPcom 192.168.1.68 .1.3.6.1.2.1.1.4.0 s "equipo12@gmail.com"
SNMPv2-MIB::sysContact.0 = STRING: equipo12@gmail.com
    
```

Figura 44: Ejercicio MIB

b) Ahora realizamos de nuevo la consulta para visualizar los cambios.

```

Archivo  Editar  Ver  Buscar  Terminal  Ayuda
enrique@enriqueHP ~/Documentos/Redes3/observium
└─ snmpget -c SNMPcom -v 2c 192.168.1.68 .1.3.6.1.2.1.1.4.0
SNMPv2-MIB::sysContact.0 = STRING: equipo12@gmail.com
    
```

Figura 45: Ejercicio MIB

12. Dibuja la MIB del agente.

a) **Linux**

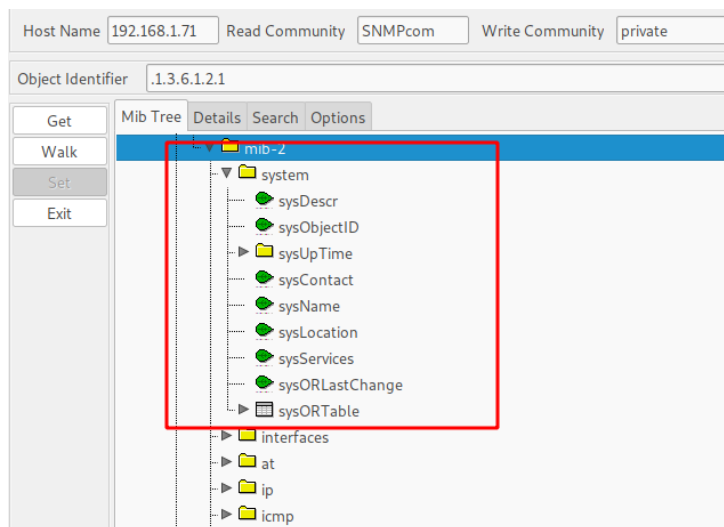


Figura 46: Ejercicio MIB

b) **Windows**

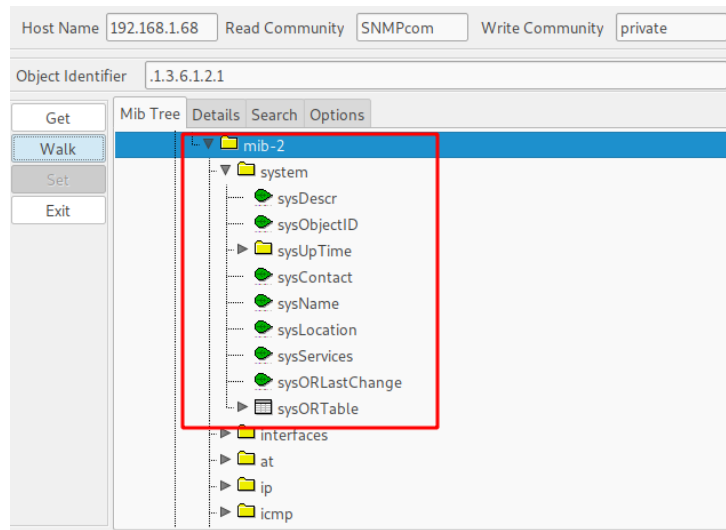


Figura 47: Ejercicio MIB

3.2. Análisis de tráfico

Para esta sección utilizaremos la herramienta Wireshark, el primer comando será una petición para obtener el contacto.

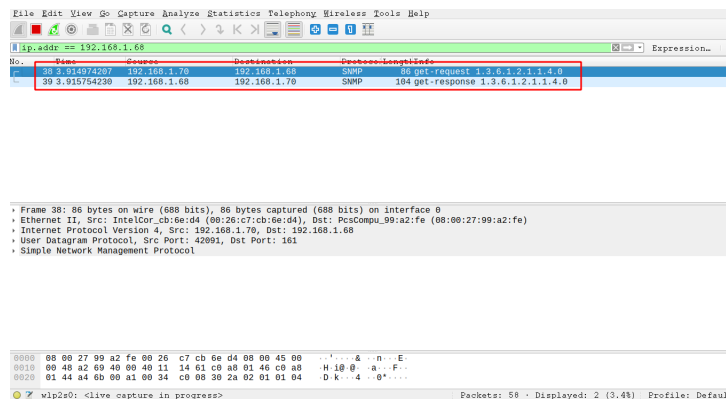


Figura 48: Análisis de tráfico

Ahora aplicamos un **set**.

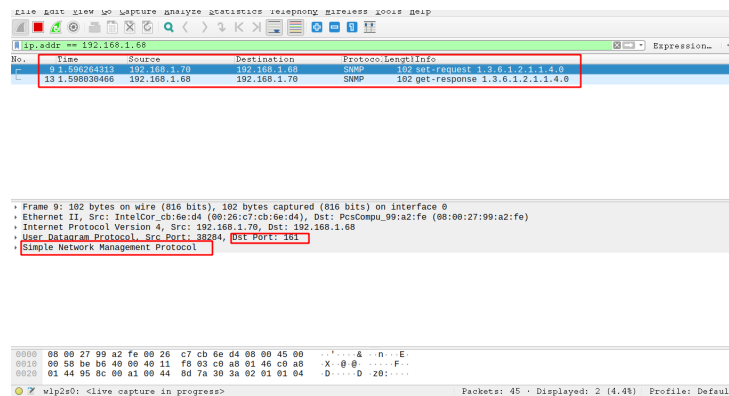


Figura 49: Análisis de tráfico

Podemos observar datos interesantes como la versión y la comunidad, también el puerto al que va dirigido el mensaje y la IP destino.

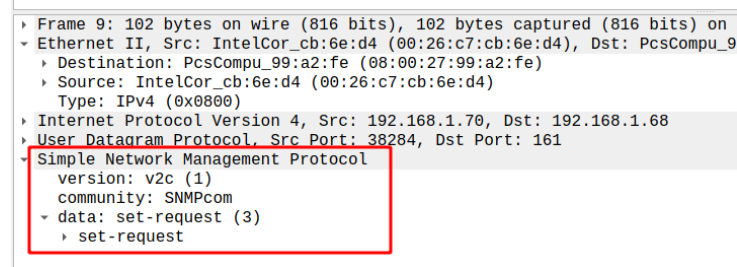


Figura 50: Análisis de tráfico

El comando **get next** obtiene el siguiente OID.

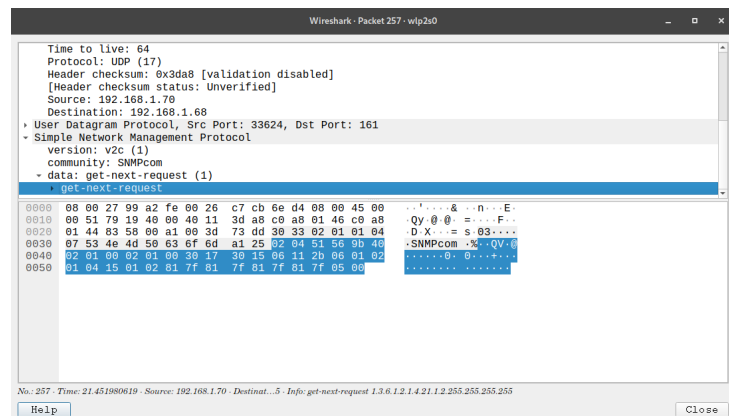


Figura 51: Análisis de tráfico

Ahora usamos el comando **snmpwalk**.

```

Archivo  Editar  Ver  Buscar  Terminal  Ayuda
->
enrique@enriqueHP: ~/Documentos/Pedec3/observium
-> snmpwalk -c SNMPcom -v 2c 192.168.1.68 .1.3.6.1.2.1.4.21
P-MIB::1.3.6.1.2.1.4.21.0.0.0 = IPAddress: 0.0.0.0
P-MIB::1.3.6.1.2.1.4.21.1.1.127.0.0.0 = IPAddress: 127.0.0.0
P-MIB::1.3.6.1.2.1.4.21.1.1.192.168.1.0 = IPAddress: 192.168.1.0
P-MIB::1.3.6.1.2.1.4.21.1.1.192.168.1.68 = IPAddress: 192.168.1.68
P-MIB::1.3.6.1.2.1.4.21.1.1.192.168.1.255 = IPAddress: 192.168.1.255
P-MIB::1.3.6.1.2.1.4.21.1.1.224.0.0.0 = IPAddress: 224.0.0.0
P-MIB::1.3.6.1.2.1.4.21.1.1.255.255.255.255 = IPAddress: 255.255.255.255

```

Figura 52: Análisis de tráfico

Analizando la información en WireShark notamos que este comando aplica el comando **get-next** para poder ir avanzando entre todos los OID, este es un excelente comando para ir navegando entre ellas así como en los directorios.

No.	Time	Source	Destination	Protocol	Length	Info
229	21.443192263	192.168.1.70	192.168.1.68	SNMP	85	get-next-request 1.3.6.1.2.1.4.21
230	21.444544003	192.168.1.68	192.168.1.70	SNMP	95	get-response 1.3.6.1.2.1.4.21.1.1.0.0.0.0
231	21.444772538	192.168.1.70	192.168.1.68	SNMP	91	get-next-request 1.3.6.1.2.1.4.21.1.1.0.0.0.0
232	21.445131801	192.168.1.68	192.168.1.70	SNMP	95	get-response 1.3.6.1.2.1.4.21.1.1.127.0.0.0
233	21.445284027	192.168.1.70	192.168.1.68	SNMP	91	get-next-request 1.3.6.1.2.1.4.21.1.1.127.0.0.0
235	21.445855922	192.168.1.70	192.168.1.68	SNMP	93	get-next-request 1.3.6.1.2.1.4.21.1.1.192.168.1.0
236	21.446172320	192.168.1.68	192.168.1.70	SNMP	97	get-response 1.3.6.1.2.1.4.21.1.1.192.168.1.0
237	21.446340326	192.168.1.70	192.168.1.68	SNMP	93	get-next-request 1.3.6.1.2.1.4.21.1.1.192.168.1.68
238	21.446824635	192.168.1.68	192.168.1.70	SNMP	98	get-response 1.3.6.1.2.1.4.21.1.1.192.168.1.255
239	21.447015545	192.168.1.70	192.168.1.68	SNMP	94	get-next-request 1.3.6.1.2.1.4.21.1.1.192.168.1.255
240	21.447410739	192.168.1.68	192.168.1.70	SNMP	96	get-response 1.3.6.1.2.1.4.21.1.1.224.0.0.0
241	21.447570800	192.168.1.70	192.168.1.68	SNMP	92	get-next-request 1.3.6.1.2.1.4.21.1.1.224.0.0.0
242	21.447873840	192.168.1.68	192.168.1.70	SNMP	99	get-response 1.3.6.1.2.1.4.21.1.1.255.255.255.255
243	21.448103326	192.168.1.70	192.168.1.68	SNMP	95	get-next-request 1.3.6.1.2.1.4.21.1.1.255.255.255.255
244	21.448354141	192.168.1.68	192.168.1.70	SNMP	94	get-response 1.3.6.1.2.1.4.21.1.2.0.0.0.0
245	21.448733925	192.168.1.70	192.168.1.68	SNMP	91	get-next-request 1.3.6.1.2.1.4.21.1.2.0.0.0.0
246	21.449066208	192.168.1.68	192.168.1.70	SNMP	92	get-response 1.3.6.1.2.1.4.21.1.2.127.0.0.0
247	21.449218396	192.168.1.70	192.168.1.68	SNMP	91	get-next-request 1.3.6.1.2.1.4.21.1.2.127.0.0.0
248	21.449589888	192.168.1.68	192.168.1.70	SNMP	96	get-response 1.3.6.1.2.1.4.21.1.2.192.168.1.0
249	21.449745848	192.168.1.70	192.168.1.68	SNMP	93	get-next-request 1.3.6.1.2.1.4.21.1.2.192.168.1.0
250	21.450155248	192.168.1.68	192.168.1.70	SNMP	94	get-response 1.3.6.1.2.1.4.21.1.2.192.168.1.68
251	21.450417511	192.168.1.70	192.168.1.68	SNMP	93	get-next-request 1.3.6.1.2.1.4.21.1.2.192.168.1.68
252	21.450791032	192.168.1.68	192.168.1.70	SNMP	97	get-response 1.3.6.1.2.1.4.21.1.2.192.168.1.255
253	21.450942253	192.168.1.70	192.168.1.68	SNMP	94	get-next-request 1.3.6.1.2.1.4.21.1.2.192.168.1.255
254	21.451309595	192.168.1.68	192.168.1.70	SNMP	95	get-response 1.3.6.1.2.1.4.21.1.2.224.0.0.0
255	21.451404010	192.168.1.70	192.168.1.68	SNMP	92	get-next-request 1.3.6.1.2.1.4.21.1.2.224.0.0.0
256	21.451830026	192.168.1.68	192.168.1.70	SNMP	98	get-response 1.3.6.1.2.1.4.21.1.2.255.255.255.255
257	21.451980610	192.168.1.70	192.168.1.68	SNMP	95	get-next-request 1.3.6.1.2.1.4.21.1.2.255.255.255.255

Figura 53: Análisis de tráfico

4. Implementación de Modelo

En este apartado, desarrollamos una interfaz en la cual podemos agregar agentes, así como eliminarlos, también podremos ver su estado actual y principalmente, podremos consultar las MIBs.

4.1. Agregar Agente

Para agregar un agente es necesario llenar los campos que se piden en el formulario, una vez completados éstos, se recibe un mensaje de confirmación de que el agente fue agregado exitosamente.

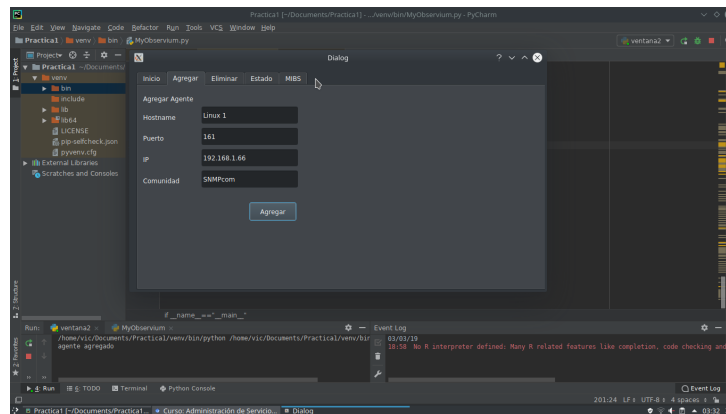


Figura 54: Interfaz

4.2. Eliminar Agente

Para esta ventana utilizaremos el HostName para traer los datos del agente para eliminarlo, así como sus datos correspondientes.

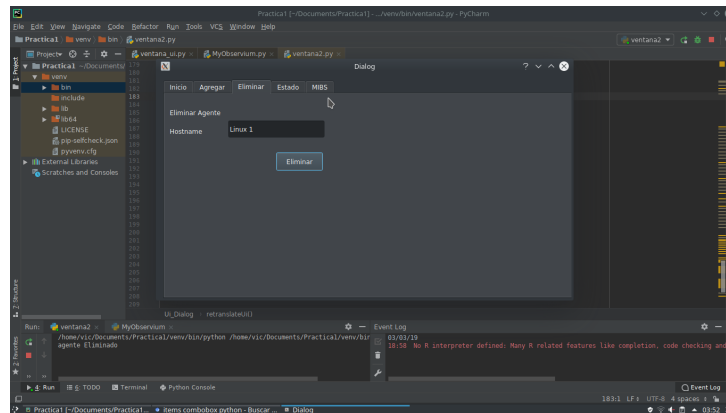


Figura 55: Interfaz

4.3. Estado del Agente

Este punto muestra algunos de los datos solicitados en la práctica, como nombre, estado y puerto en el que trabaja.

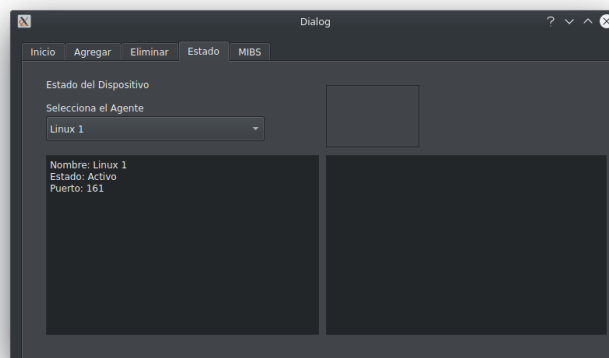


Figura 56: Interfaz

5. Conclusiones

Aquí se encuentran las conclusiones de cada miembro del equipo.

- Jimena: En lo personal esta práctica me costó trabajo pues al principio no entendía nada sobre el protocolo SNMP o su gestión, conforme fue pasando el tiempo fui empapándome más del tema hasta que logré comprenderlo. Me parece una manera muy interesante de gestionar las redes, pues normalmente sin algún gestor no nos interesamos en saber en qué puerto está trabajando, por ejemplo, en cambio ahora podemos conocer esa y mucha más información sobre los agentes que tengamos disponibles en nuestra computadora. En general el protocolo SNMP es bastante amplio y tiene un conjunto de comandos muy útiles para cualquier acción que queramos realizar ya sea en cuestión a los agentes o a las MIBs, las cuales, por cierto, me agradaron mucho, son consultas muy interesantes que tu vas formando conforme a la información que quieres obtener.
- Víctor: En esta práctica es importante mencionar la dificultad de programar varias de las funciones mostradas anteriormente, sin embargo conocer más a detalle los procesos y lo que es SNMP en general es una buena experiencia, ya que tiene una gran complejidad y una variedad de funciones muy importantes, así mismo se puso en práctica el conocimiento sobre las MIB y OID explicados en la clase.
- Enrique: En esta práctica se vieron los principios para poder realizar un monitoreo de un equipo en la red y cómo comunicarnos, tanto para poder obtener y cambiar propiedades del sistema monitorizado, en particular éste gracias al protocolo de comunicación SNMP. Aprendimos a configurar el servicio SNMP tanto en Windows como en Linux, en particular la instalación en Windows fue más rápida que la instalación en Linux, pues teníamos que configurar un archivo, mientras que en Windows lo modificábamos sobre una interfaz gráfica.
Por otro lado conocer que existen herramientas para poder monitorizar nuestros dispositivos en red, fue algo que me interesó ya que puedo monitorizar las máquinas de una empresa, sin importar desde donde me encuentre. Algo que al inicio me costó fueron los OID, ya que al inicio no le entendía al árbol de la MIB, pero conforme realizaba los ejercicios de la práctica fui entendiendo mejor los OID, por supuesto el comando que más me gustó de SNMP fue el comando `snmpwalk` ya que con este comando podía visualizar la información de las MIBs al mismo tiempo que consultaba los comandos en Wireshark.

Para la tercera parte de la práctica programar nuestro propio Observium, fue algo tardado pero la parte que más me gustó fue programar las consultas para obtener los valores de los OID requeridos, esto a mi parecer fue fácil ya que existe una API en python que nos facilitaba este trabajo, por otro lado la parte más tediosa o un poco complicada fue la interfaz gráfica, ya que ninguno de mis compañeros es bueno realizando interfaces, por este motivo el desarrollo de esta parte de la práctica fue un poco tedioso. En general creo que se cumplieron los propósitos de la práctica el cual fue conocer el protocolo SNMP y la monitorización de servicios en red.

6. Referencias Bibliográficas

- Briceño Caryuly, Rosales, Protocolo snmp (protocolo sencillo de administración de redes) . Télématique [en línea] 2004, 3 (enero - junio) : [Fecha de consulta: 4 de marzo de 2019] Disponible en: <http://www.redalyc.org/articulo.oa?id=ISSN 1856-4194>
- Guerrero David, Protocolo SNMP. Linux Journal[en línea] 1998: [Fecha de consulta; 4 de marzo de 2019] Disponible en: <http://redesdecomputadores.umh.es/aplicac>