



INSTITUTO POLITÉCNICO NACIONAL  
ESCUELA SUPERIOR DE CÓMPUTO

## Práctica 1

*Integrantes:*

López Ayala Eric Alejandro  
López Romero Joel

*Profesora:*

Pérez de Los Santos Mondragón  
Tanibet

*Grupo:*

4CM1

*Asignatura:*

Administración de Servicios en Red

# Índice general

<b>1. Introducción teórica</b>	<b>1</b>
1.1. SNMP . . . . .	1
1.2. MIB . . . . .	2
1.2.1. MIB: Descripción . . . . .	3
1.2.2. Tipos de nodos . . . . .	3
1.2.3. Nodos estructurales . . . . .	4
<b>2. Desarrollo.</b>	<b>5</b>
2.1. Arquitectura Básica del SNMP . . . . .	5
2.1.1. Instalación y configuración de la máquina virtual usando VirtualBox . . . . .	5
2.1.2. Instalación y Configuración de la Estación de Gestión . . . . .	7
2.1.3. Instalación y configuración del agente de gestión en el sistema operativo Windows . . . . .	15
2.2. Resultados . . . . .	25
2.3. Implementación de un modelo (versión 1) de administración de red de SNMP . . . . .	26
2.4. Cuestionario . . . . .	31
2.4.1. Vigilancia y control de los agentes de gestión . . . . .	31
2.5. Análisis de tráfico . . . . .	44
2.5.1. Captura de paquetes SNMP formados por el comando snmpget . . . . .	44
2.5.2. Captura de paquetes SNMP formados por el comando snmpset . . . . .	45
2.5.3. Captura de paquetes SNMP formados por el comando snmpgetnext . . . . .	46
2.5.4. Captura de paquetes SNMP formados por el comando snmptable . . . . .	47
2.5.5. Captura de paquetes SNMP formados por el comando snmpwalk . . . . .	48
2.6. Conclusiones . . . . .	49
2.6.1. Conclusión - López Ayala Eric Alejandro . . . . .	49
2.6.2. Conclusión - López Romero Joel . . . . .	49
Referencias . . . . .	50

# 1 | Introducción teórica

## 1.1. SNMP

**SNMP** (*Simple Network Management Protocol*) es el protocolo más utilizado para la gestión de redes IP basadas en internet. El protocolo simple de administración en red (SNMP) fue publicado en 1988, y fue diseñado para proporcionar una implementación sencilla, así como facilitar el trabajo de gestión de redes de múltiples proveedores, debido a las grandes dimensiones que estaba tomando empleación de los servicios ofrecidos por TCP/IP. La especificación SNMP tiene como objetivo:

- Definir un protocolo para el intercambio de información entre uno o más sistemas de gestión y un número de agentes.
- Proporcionar un marco para dar formato y almacenamiento de información de gestión.
- Define una serie de variables de información de gestión de propósito general.

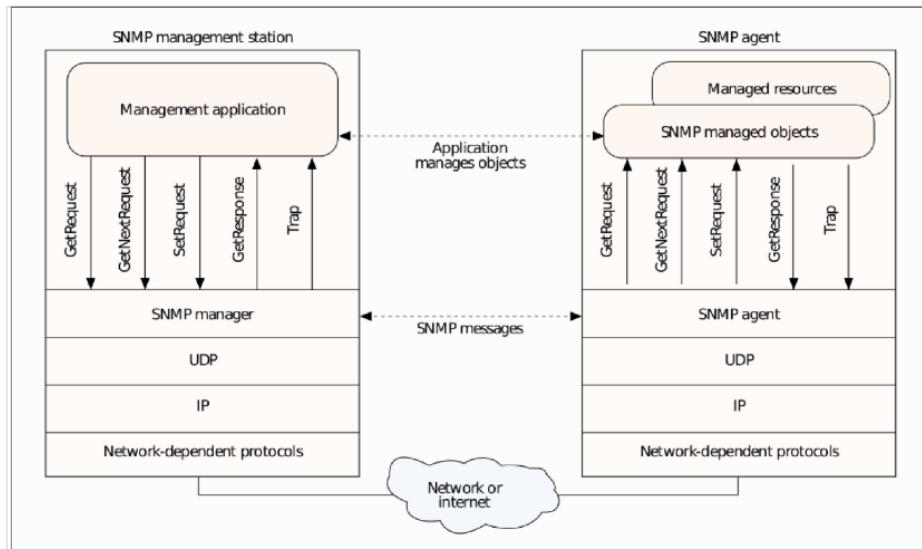


Figura 1.1: Arquitectura SNMP

SNMP surge a raíz del interés mostrado por la IAB (*Internet Activities Board*) en encontrar un protocolo de gestión que fuese válido para la red Internet. De los tres grupos de trabajo que inicialmente se formaron, finalmente el SNMP (RFC 1098) fue el adoptado, incluyendo éste algunos de los aspectos más relevantes presentados por los otros dos: HEMS (*High-Level Management System*) y SGMP (*Simple Gateway Monitoring Protocol*). [1]

Para el protocolo SNMP la red constituye un conjunto de elementos básicos:

- **Administradores o Management Stations:** Ubicados en el/los equipo/s de gestión de red y Gestores.
- **Network Agents:** Elementos pasivos ubicados en los nodos -host, routers, modems, multiplexores, etc.- a ser gestionados).

Siendo los segundos los que envían información a los primeros, relativa a los elementos gestionados, por iniciativa propia o al ser interrogados (polling) de manera secuencial, apoyándose en los parámetros contenidos en sus MIB (*Management Information Base*). A través del MIB se tiene acceso a la información para la gestión, contenida en la memoria interna del dispositivo en cuestión. La MIB es una base de datos completa y bien definida, con una estructura en árbol adecuada para manejar diversos grupos de objetos (información sobre variables/valores que se pueden adoptar), con identificadores exclusivos para cada objeto. [2]

## 1.2. MIB

La Base de Información Gestionada (**Management Information Base o MIB**) es un tipo de base de datos que contiene información jerárquica, estructurada en forma de árbol, de todos los parámetros gestionables en cada dispositivo gestionado de una red de comunicaciones.

Es parte de la gestión de red definida en el modelo **OSI**, define las variables usadas por el protocolo **SNMP** para supervisar y controlar los componentes de una red. Está compuesta por una serie de objetos que representan los dispositivos (como enrutadores y commutadores) en la red. Cada objeto manejado en un **MIB** tiene un identificador de objeto único e incluye el tipo de objeto (tal como contador, secuencia o gauge), el nivel de acceso (tal como lectura y escritura), restricciones de tamaño, y la información del rango del objeto.

### 1.2.1. MIB: Descripción

- Las MIB tienen un formato común de forma que aun cuando los dispositivos sean de fabricantes distintos, puedan ser administrados con un protocolo muy general.
- **Protocolo de administración:** es el protocolo mediante el cual se consultan los objetos administrados enviando la información a la estación administradora.
- Las MIB suelen ser modificadas cada cierto tiempo para añadir nuevas funcionalidades, eliminar ambigüedades y arreglar fallos. Estos cambios se han de hacer de acuerdo con la sección 10 del RFC 2578.

### 1.2.2. Tipos de nodos

Existen dos tipos de nodos: **estructurales y de información**.

- **Nodos estructurales:** sólo tienen descrita su posición en el árbol. Son "ramas". Por ejemplo:  
*ip OBJECT IDENTIFIER ::= 1.3.6.1.2.1.4*
- **Nodos de información:** son nodos "hoja". De ellos no cuelga ningún otro nodo. Estos nodos están basados en la macro OBJECT TYPE, por ejemplo:
  - ipInReceives OBJECT TYPE SYNTAX Counter
  - ACCESS read-only
  - STATUS mandatory
  - DESCRIPTION "texto descriptivo indicando para qué vale"
  - ::= ip 3 Este fragmento ASN.1 nos indica que el objeto ipInReceives.<sup>es</sup> un contador de sólo lectura que es obligatorio incorporar si se quiere ser compatible con la MIB-II (aunque luego no se utilice) y que cuelga del nodo ip con valor tres. Como antes hemos visto el nodo estructural ip con su valor absoluto, podemos ver que identificador de objeto de ipInReceives.<sup>es</sup> "1.3.6.1.2.1.4.3".[3]

### 1.2.3. Nodos estructurales

- **SYSTEM:** De este nodo se encuentran objetos que proporcionan información genérica del sistema gestionado. Por ejemplo, dónde se encuentra el sistema, quién lo administra, etc.
- **INTERFACES:** En este grupo está la información de los interfaces de red presentes en el sistema. Incorpora estadísticas de los eventos ocurridos en el mismo.
- **AT:** Este nodo es obsoleto, pero se mantiene para preservar la compatibilidad con la MIB-I. En él se almacenan las direcciones de nivel de enlace correspondientes a una dirección IP.
- **IP:** En este grupo se almacena la información relativa a la capa IP, tanto de configuración como de estadísticas.
- **ICMP:** En este nodo se almacenan contadores de los paquetes ICMP entrantes y salientes.
- **TCP:** En este grupo está la información relativa a la configuración, estadísticas y estado actual del protocolo TCP.
- **UDP:** En este nodo está la información relativa a la configuración, estadísticas del protocolo UDP.
- **EGP:** Aquí está agrupada la información relativa a la configuración y operación del protocolo EGP.
- **TRANSMISSION:** De este nodo cuelgan grupos referidos a las distintas tecnologías del nivel de enlace implementadas en las interfaces de red del sistema gestionado. [4]

## 2 | Desarrollo.

### 2.1. Arquitectura Básica del SNMP

Durante esta parte del desarrollo de la práctica, se realizará la implementación de la arquitectura básica SNMP. El modelo de administración de red que utiliza SNMP incluye los siguientes elementos:

- Estación de gestión.
- Agente de gestión.
- Base de información de gestión.
- Protocolo de gestión de redes.

A continuación se enlistan las tareas para realizar la instalación y configuración de dichos elementos.

#### 2.1.1. Instalación y configuración de la máquina virtual usando VirtualBox

Para poder implementar la arquitectura básica de SNMP se usará una máquina virtual, esto con la finalidad de poder instalar y configurar los distintos agentes dentro la arquitectura, ya que estos poseen distintos sistemas operativos.

**Oracle VM VirtualBox es un software de virtualización** para arquitecturas x86/amd64. Por medio de esta aplicación es posible instalar sistemas operativos adicionales, conocidos como «sistemas invitados», dentro de otro sistema operativo «anfitrión», cada uno con su propio ambiente virtual. Entre los sistemas operativos soportados (en modo anfitrión) se encuentran GNU/Linux, Mac OS X, OS/2 Warp , Microsoft Windows, y Solaris/OpenSolaris, y dentro de ellos es posible virtualizar los sistemas operativos FreeBSD, GNU/Linux, OpenBSD, OS/2 Warp, Windows, Solaris, MS-DOS y muchos otros.

*Nota: El siguiente procedimiento describe la instalación paso a paso de VirtualBox para el sistema operativo Linux - Ubuntu 16.04 Xenial.*

1. Abrir la terminal y usar el comando *apt-add-repository*, para agregar el enlace del repositorio al archivo */etc/apt/sources.list*

```
1 sudo apt-add-repository  
2 "deb http://download.virtualbox.org/virtualbox/debian  
3 \$(lsb_release -sc) contrib\"
```

2. Agregar la llave de seguridad

```
1 wget -q https://www.virtualbox.org/download/oracle_vbox_2016.asc -O-  
2 | sudo apt-key add -
```

3. Instalar VirtualBox

```
1 sudo apt-get update  
2 sudo apt-get install virtualbox-5.2
```

4. Reiniciar la computadora

5. Ejecutar VirtualBox mediante la terminal

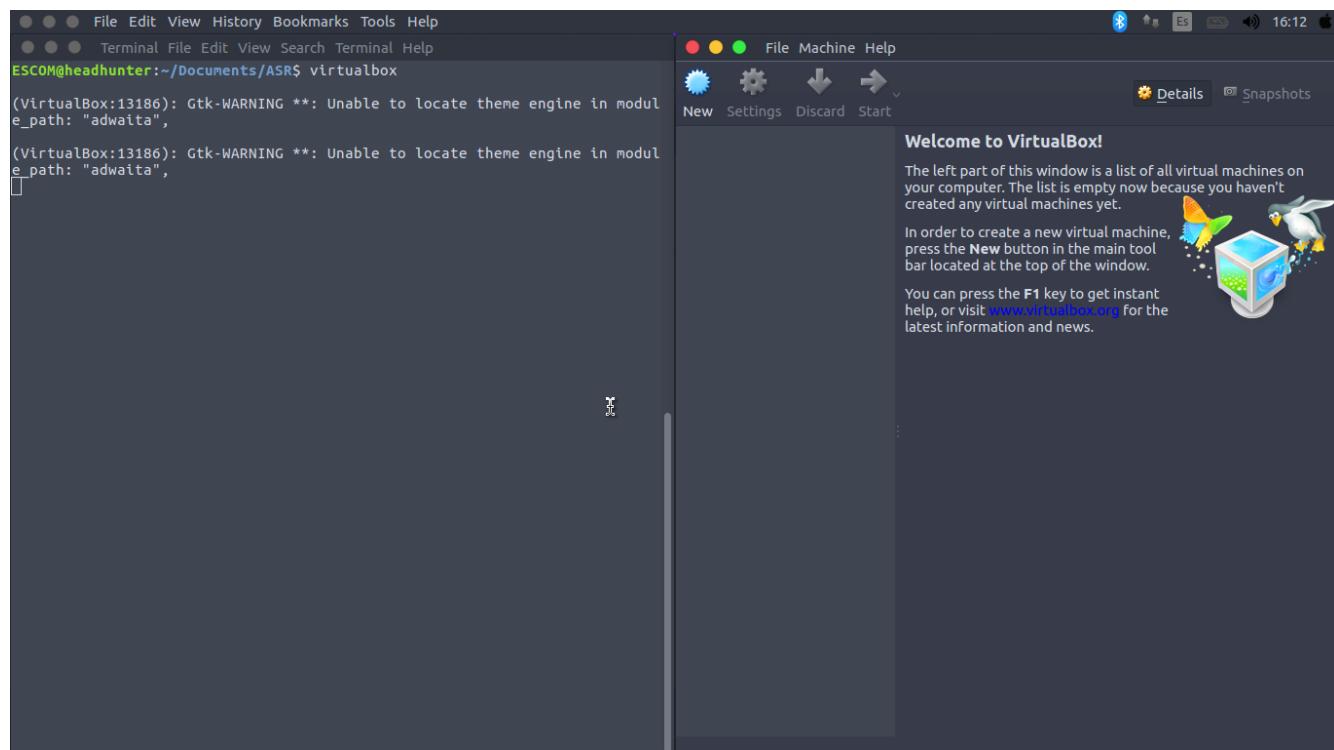


Figura 2.1: Ejecutando VirtualBox

## 2.1.2. Instalación y Configuración de la Estación de Gestión

Para la estación de gestión se hizo uso de Observium. Observium es un sistema operativo dedicado a la gestión y monitoreo de red basado en SNMP. Esta plataforma fue implementada en PHP e incluye un soporta a un amplio rango de hardware de red y sistemas operativos incluyendo Cisco, Windows, Linux, HP, Dell, FreeBSD, Juniper, Brocade, Netscaler, NetApp y otras.

### Instalación de Observium en una máquina virtual

1. Para instalar Observium, se descarga la imagen ISO del sitio web: <https://www.turnkeylinux.org/observium>, configurando una máquina virtual en VirtualBox.



Figura 2.2: ISO de Observium.

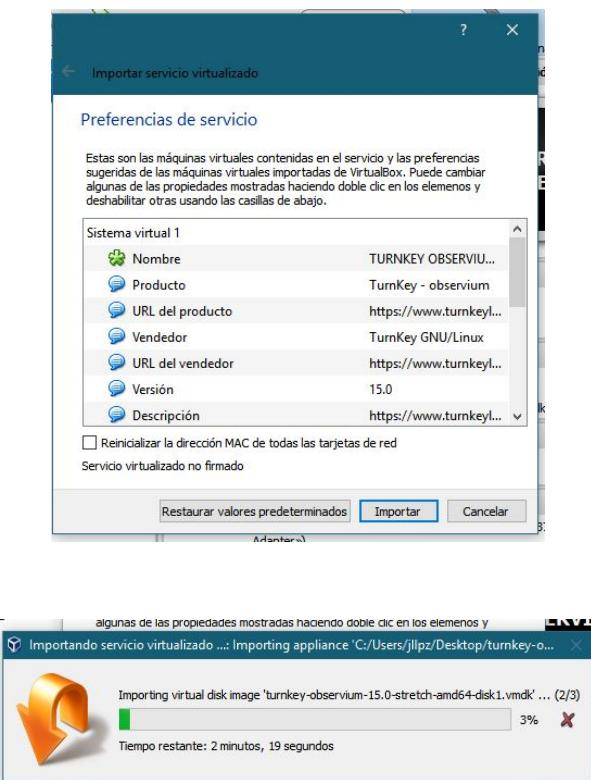


Figura 2.3: Importación de Observium.

2. Una vez que importamos Observium, ya está listo para instalarse:

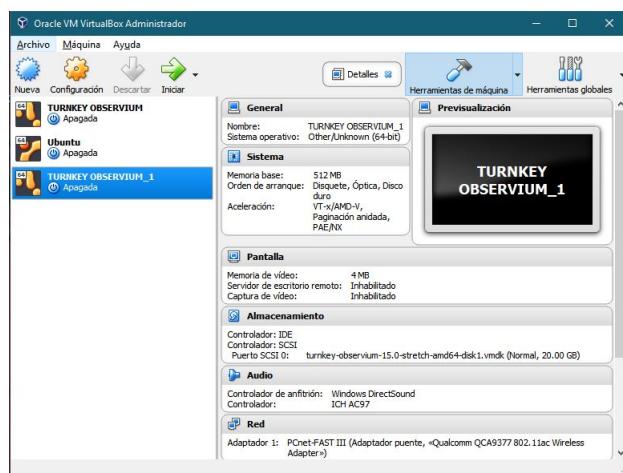


Figura 2.4: Ejecutando Observium.

3. Configuración básica del sistema operativo Observium:

- 3.1. Elección del modo de instalación del sistema operativo.
- 3.2. Selección del método de partición.
- 3.3. Confirmación de para iniciar el proceso de partición del disco duro.
- 3.4. Instalación del GRUB del sistema operativo.
- 3.5. Reinicio de la máquina virtual.

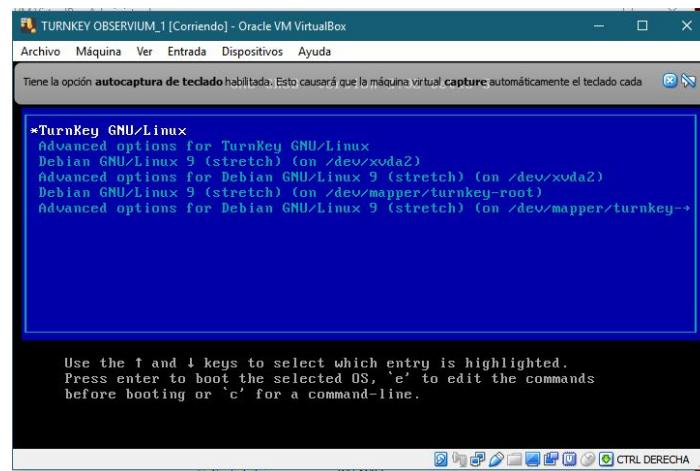


Figura 2.5: Instalación de Observium.

4. Creamos la cuenta *root*, para ello ingresamos el password que tendrá el usuario root, en este caso es: **Jllpzrmr<7**. Después realizamos la confirmación de dicha contraseña.

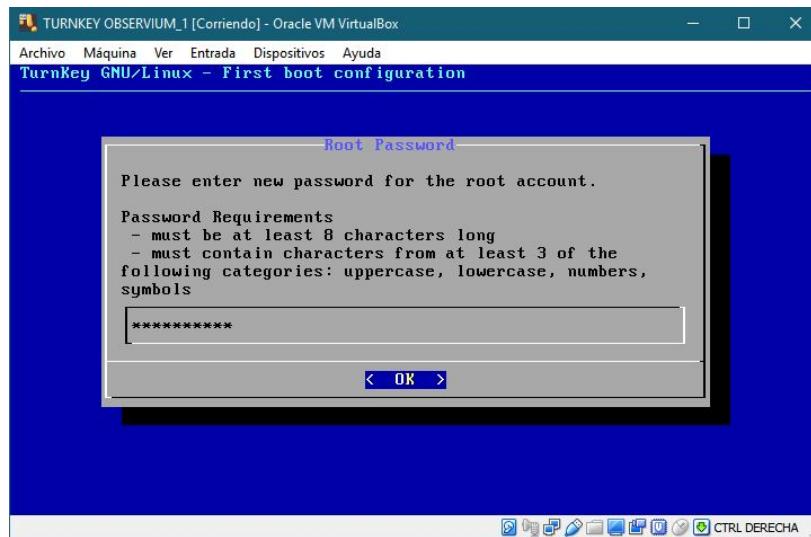


Figura 2.6: Creación del usuario Root.

5. Creamos la cuenta *mysql*, para ello ingresamos el password que tendrá el usuario MySQL, en este caso es: **Jllpzrmr<7**. Después realizamos la confirmación de dicha contraseña.

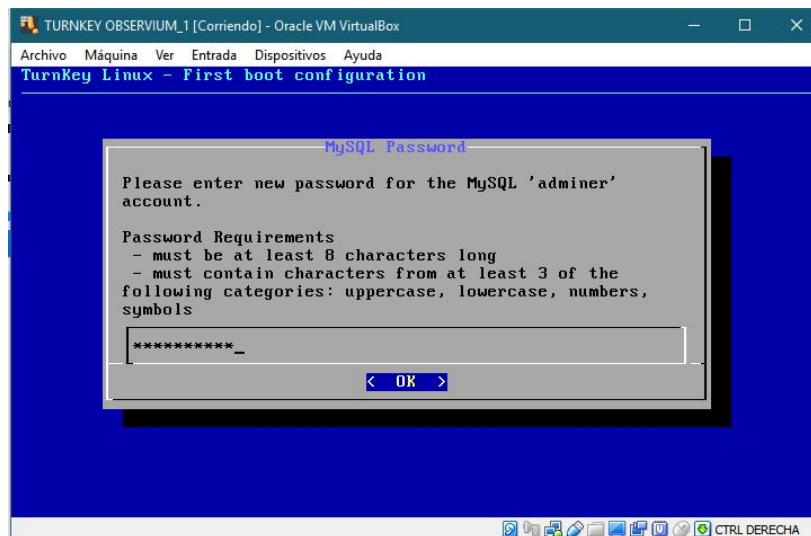


Figura 2.7: Creación del usuario MySQL.

6. Creamos la cuenta *admin*, para ello ingresamos el password que tendrá el usuario admin, en este caso es: **Jllpzr<sub>m</sub>r<7**. Despu s realizamos la confirmaci n de dicha contrase a.

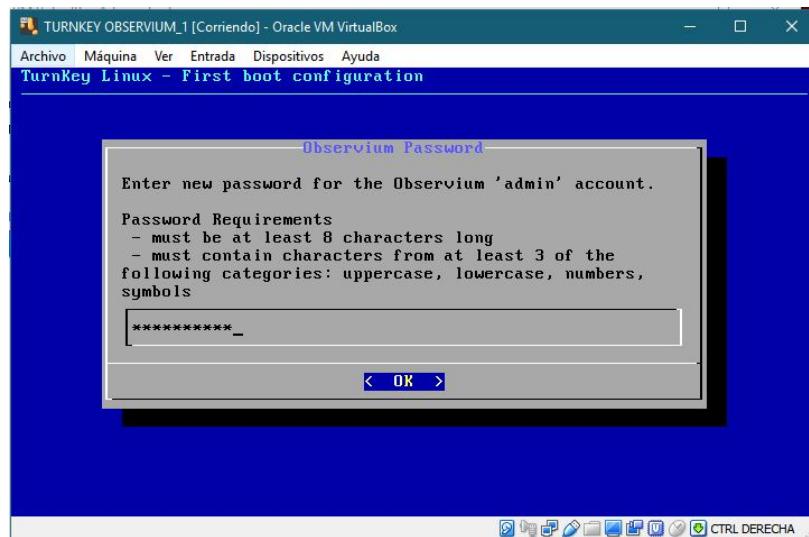


Figura 2.8: Creaci n del usuario Admin.

7. Nos saltamos la inicializaci n de los servicios HUB.

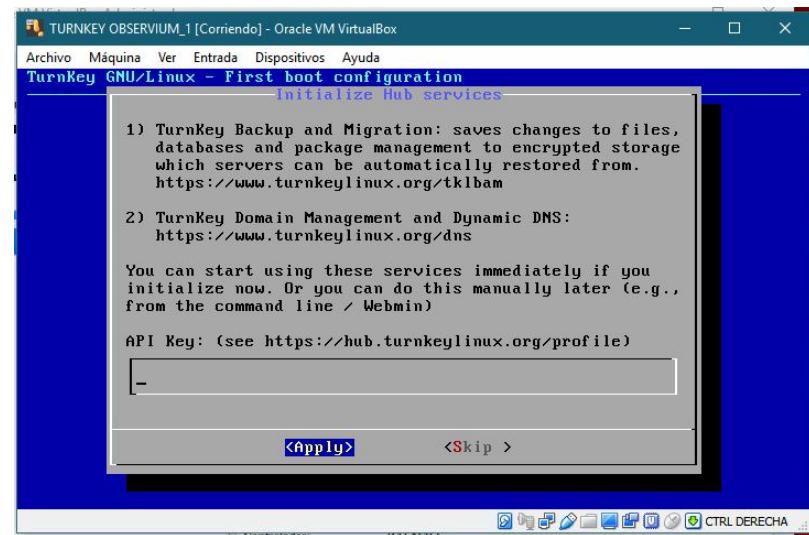


Figura 2.9: Servicios HUB.

8. Agregamos un correo de contacto.

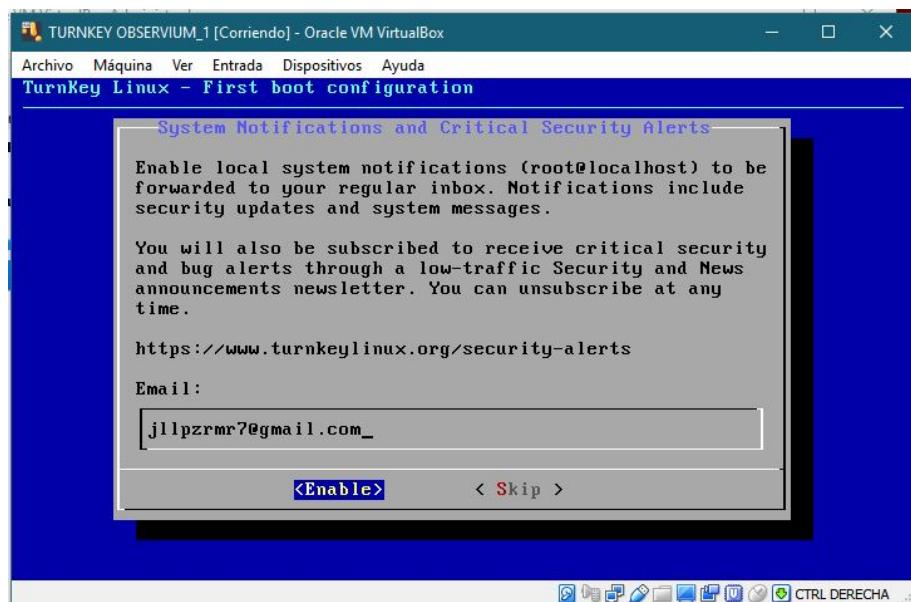
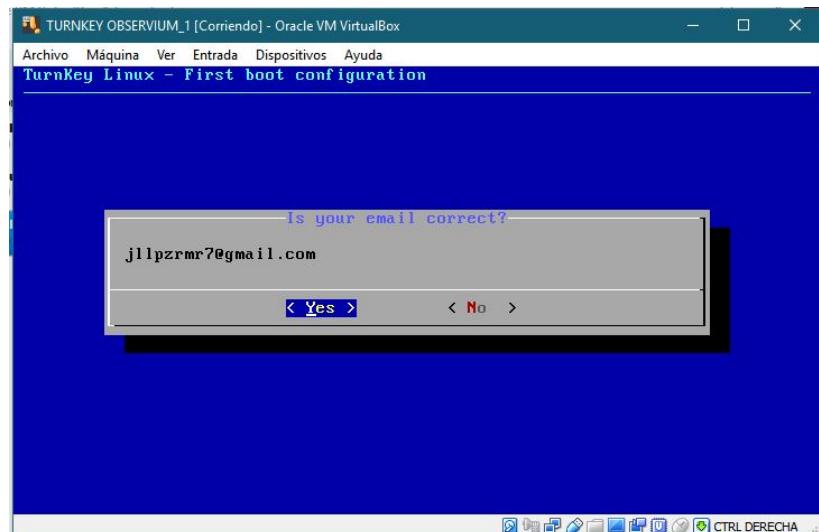


Figura 2.10: Correo de contacto.

9. Confirmamos el correo de contacto.



Confirmación de correo de contacto.

10. Nos saltamos las actualizaciones de seguridad de Turnkey.

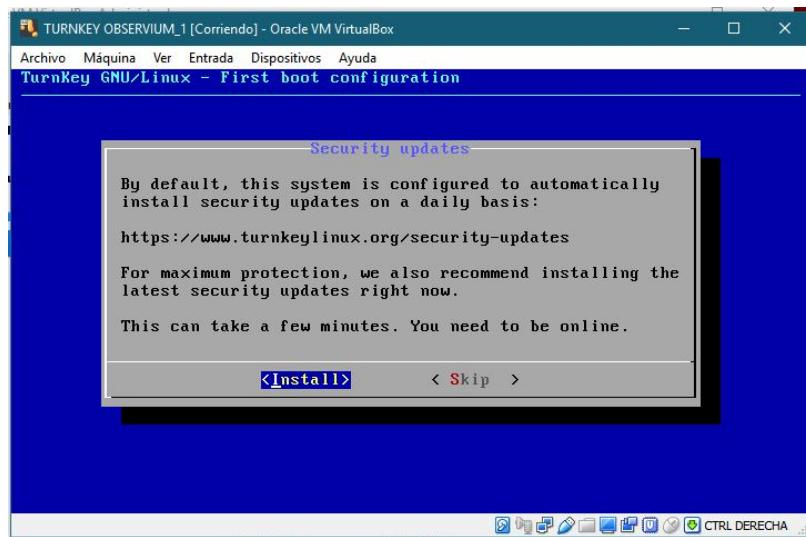


Figura 2.11: Salto de la instalación de las actualizaciones de seguridad.

11. Reiniciamos la máquina virtual de observium.

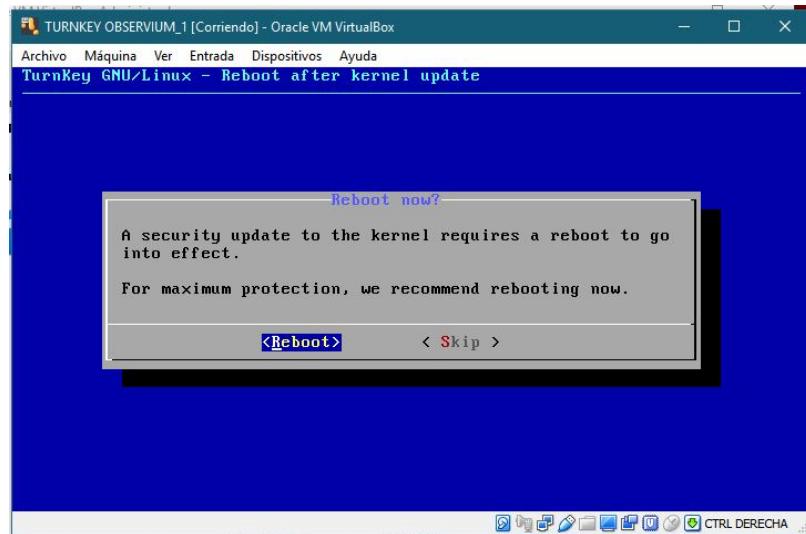


Figura 2.12: Reinicio de la máquina virtual de Observium.

12. Una vez terminada las configuraciones avanzadas de observium, se mostrarán las direcciones de red configuradas, nos aseguramos que estemos en el mismo segmento de red que nuestro sistema operativo nativo; de no ser así configurar el DHCP.

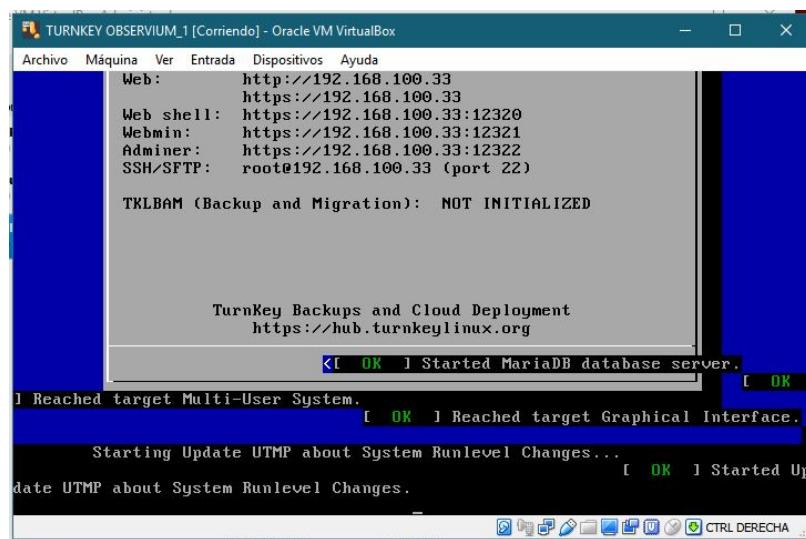


Figura 2.13: Configuración de direcciones de red.

13. Instalación completa.

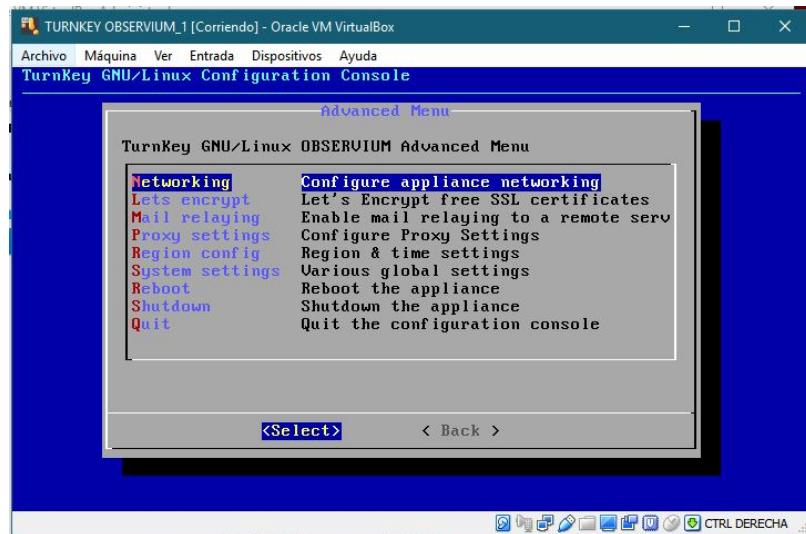


Figura 2.14: Menú de Observium, instalación completada.

## 2.1.3. Instalación y configuración del agente de gestión en el sistema operativo Windows

### Activación del protocolo SNMP en Windows

1. Agregamos la característica SNMP para Windows.



Figura 2.15: Habilitación de la característica de SNMP.

2. Activamos el servicio de captura SNMP en la configuración de servicios de Windows.

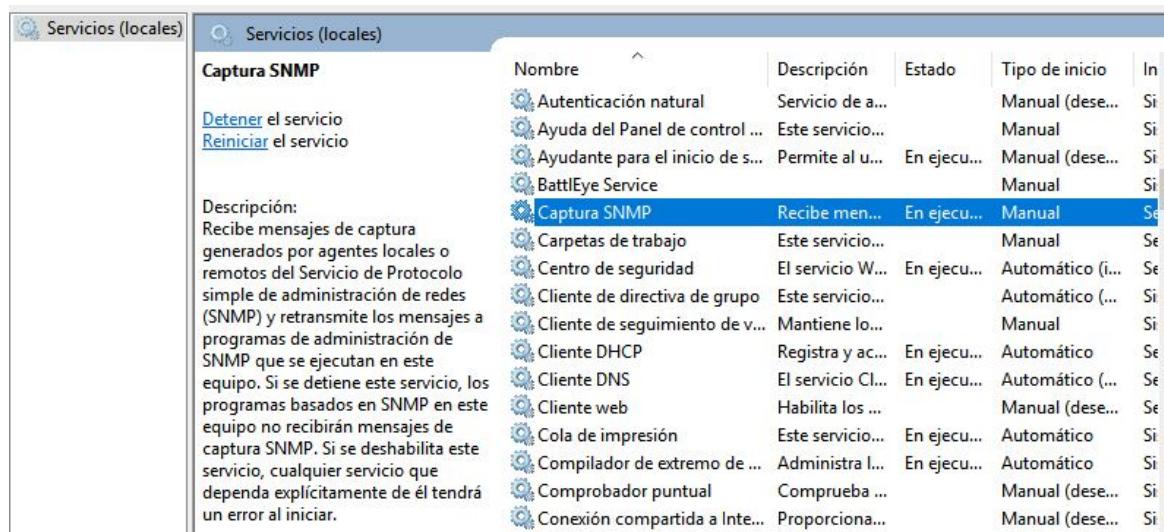
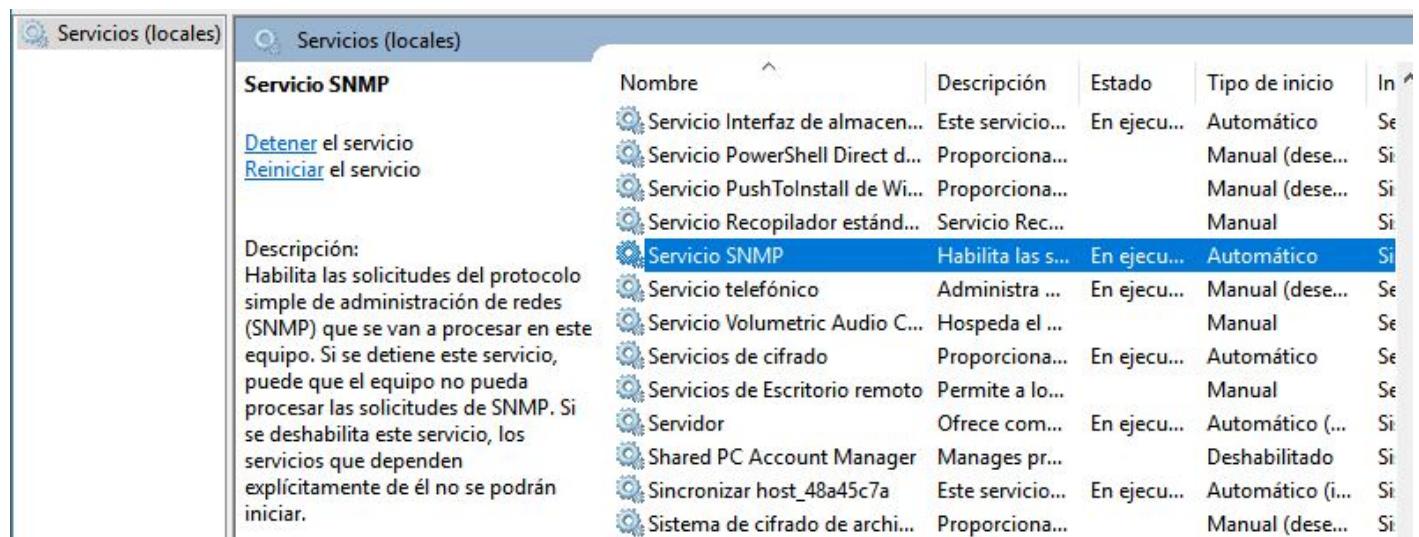


Figura 2.16: Activación el servicio captura SNMP.

3. Verificamos que se encuentre el servicio de SNMP en la configuración de servicio de Windows.



Servicio	Nombre	Descripción	Estado	Tipo de inicio	Iniciar
Detener el servicio	Servicio Interfaz de almacenamiento	Este servicio...	En ejecu...	Automático	Si
Reiniciar el servicio	Servicio PowerShell Direct d...	Proporciona...		Manual (dese...	Si
	Servicio PushToInstall de Wi...	Proporciona...		Manual (dese...	Si
	Servicio Recopilador estánd...	Servicio Rec...		Manual	Si
	<b>Servicio SNMP</b>	Habilita las s...	En ejecu...	<b>Automático</b>	Si
	Servicio telefónico	Administra ...	En ejecu...	Manual (dese...	Se
	Servicio Volumetric Audio C...	Hospeda el ...		Manual	Se
	Servicios de cifrado	Proporciona...	En ejecu...	Automático	Se
	Servicios de Escritorio remoto	Permite a lo...		Manual	Se
	Servidor	Ofrece com...	En ejecu...	Automático (...)	Si
	Shared PC Account Manager	Manages pr...		Deshabilitado	Si
	Sincronizar host_48a45c7a	Este servicio...	En ejecu...	Automático (i...	Si
	Sistema de cifrado de archi...	Proporciona...		Manual (dese...	Si

Figura 2.17: Verificación del servicio SNMP.

4. Abrimos las propiedades del servicio SNMP, en la pestaña de capturas agregar una nueva comunidad con el nombre de **comunidadSNMP**.

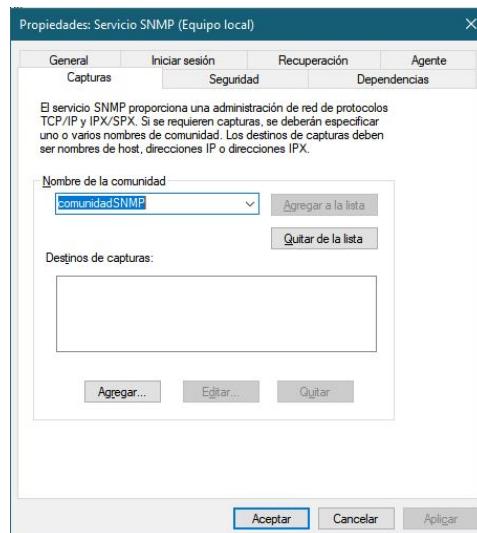


Figura 2.18: Agregando una nueva comunidad SNMP.

5. En las propiedades de servicio SNMP, habilitar la opción de recibir paquetes desde cualquier host.

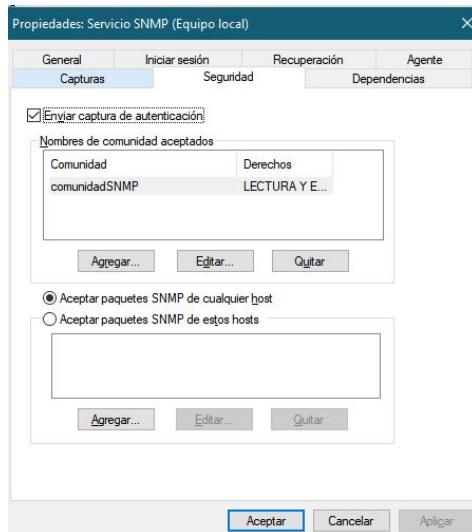


Figura 2.19: Habilitación de recepción de paquetes desde cualquier host.

6. Agregamos los permisos de lectura y escritura a la comunidad recién añadida, y damos clic en el botón aceptar.

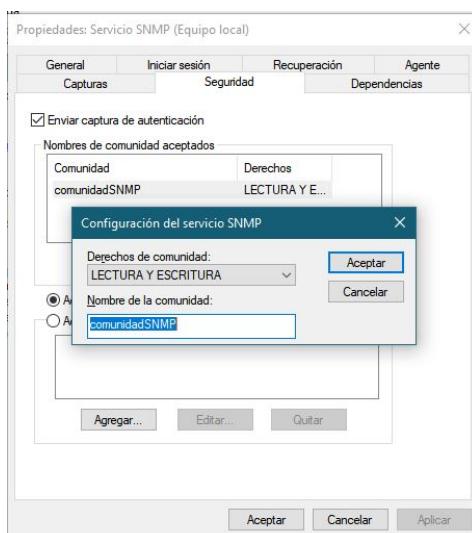
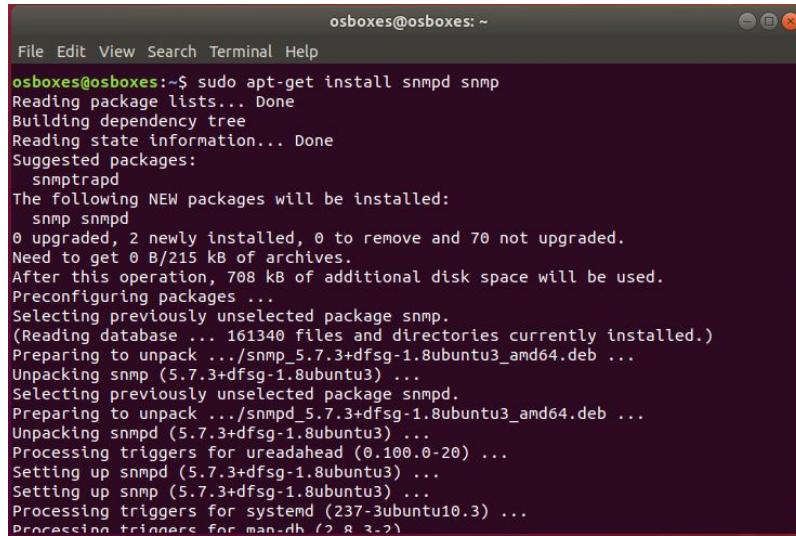


Figura 2.20: Agregando permisos R/W.

7. Finalmente reiniciamos el servicio SNMP.

## Configuración del protocolo SNMP en Linux

1. Abrimos una consola en nuestro sistema operativo linux,e instalamos NET-SNMP mediante el comando **sudo apt-get install snmpd snmp**.



```
osboxes@osboxes:~$ sudo apt-get install snmpd snmp
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  snmptrapd
The following NEW packages will be installed:
  snmp snmpd
0 upgraded, 2 newly installed, 0 to remove and 70 not upgraded.
Need to get 0 B/215 kB of archives.
After this operation, 708 kB of additional disk space will be used.
Preconfiguring packages ...
Selecting previously unselected package snmp.
(Reading database ... 161340 files and directories currently installed.)
Preparing to unpack .../snmp_5.7.3+dfsg-1.8ubuntu3_amd64.deb ...
Unpacking snmp (5.7.3+dfsg-1.8ubuntu3) ...
Selecting previously unselected package snmpd.
Preparing to unpack .../snmpd_5.7.3+dfsg-1.8ubuntu3_amd64.deb ...
Unpacking snmpd (5.7.3+dfsg-1.8ubuntu3) ...
Processing triggers for ureadahead (0.100.0-20) ...
Setting up snmpd (5.7.3+dfsg-1.8ubuntu3) ...
Setting up snmp (5.7.3+dfsg-1.8ubuntu3) ...
Processing triggers for systemd (237-3ubuntu10.3) ...
Processing triggers for man-db (2.8.3-2)
```

Figura 2.21: Instalación de NET-SNMP en Linux.

2. Una vez terminada la instalación, nos movemos a la carpeta **/etc/snmp/**, para editar el archivo **snmpd.conf**, ejecutando el script en perl mediante el comando **snmpconf -r none -g basic\_setup** configurando el archivo del siguiente modo:

- y
- laboratorio de Redes
- overlord.lae@gmail.com
- n
- y
- n
- n
- y
- Return
- Return
- n
- n
- n

```
File Edit View Search Terminal Help
root@osboxes:/home/osboxes# snmpconf -r none -g basic_setup
*****
*** Beginning basic system information setup ***
*****
Do you want to configure the information returned in the system MIB group (contact info, etc)? (default = y): y

Configuring: syslocation
Description:
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. IE, including
this token in the snmpd.conf file will disable write access to
the variable.
arguments: location_string

The location of the system: laboratorio Redes

Finished Output: syslocation "laboratorio Redes"

Configuring: syscontact
Description:
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. IE, including
this token in the snmpd.conf file will disable write access to
the variable.
arguments: contact_string

The contact information: jllpzr7@gmail.com

Finished Output: syscontact jllpzr7@gmail.com
Do you want to properly set the value of the sysServices.0 OID (if you don't know, just say no)? (default = y): n
*****
*** BEGINNING ACCESS CONTROL SETUP ***
*****
Do you want to configure the agent's access control? (default = y): y
Do you want to allow SNMPv3 read-write user based access (default = y): n
Do you want to allow SNMPv3 read-only user based access (default = y): n
Do you want to allow SNMPv1/v2c read-write community access (default = y): y

Configuring: rwcommunity
Description:
a SNMPv1/SNMPv2c read-write access community name
arguments: community [default|hostname|network/bits] [oid]

Enter the community name to add read-write access for: comunidadSNMP
The hostname or network address to accept this community name from [RETURN for a null]:
The OID that this community should be restricted to [RETURN for no-restriction]:
```

Figura 2.22: Configuración SNMP a través del archivo snmpd.conf

3. Por último reiniciamos el servicio SNMP mediante le comando **sudo service snmpd restart**.

```
The OID that this community should be restricted to [RETURN for no-restriction]:  
  
Finished Output: rwcommunity comunidadSNMP  
Do another rwcommunity line? (default = y): n  
Do you want to allow SNMPv1/v2c read-only community access (default = y): n  
*****  
*** Beginning monitoring setup ***  
*****  
Do you want to configure the agent's ability to monitor various aspects of your  
system? (default = y): n  
  
Error: An snmpd.conf file already exists in this directory.  
'overwrite', 'skip', 'rename' or 'append'? : overwrite  
  
The following files were created:  
  
snmpd.conf  
  
These files should be moved to /usr/share/snmp if you  
want them used by everyone on the system. In the future, if you add  
the -i option to the command line I'll copy them there automatically for you.  
  
Or, if you want them for your personal use only, copy them to  
/root/.snmp . In the future, if you add the -p option to the  
command line I'll copy them there automatically for you.  
  
root@osboxes:/home/osboxes# exit  
exit  
osboxes@osboxes:~$ sudo service snmpd restart  
osboxes@osboxes:~$ █
```

Figura 2.23: Reinicio del servicio SNMP.

## Publicación los agentes en Observium.

1. En Observium abrimos el archivo de agentes mediante el siguiente comando: **nano /etc/hosts**.

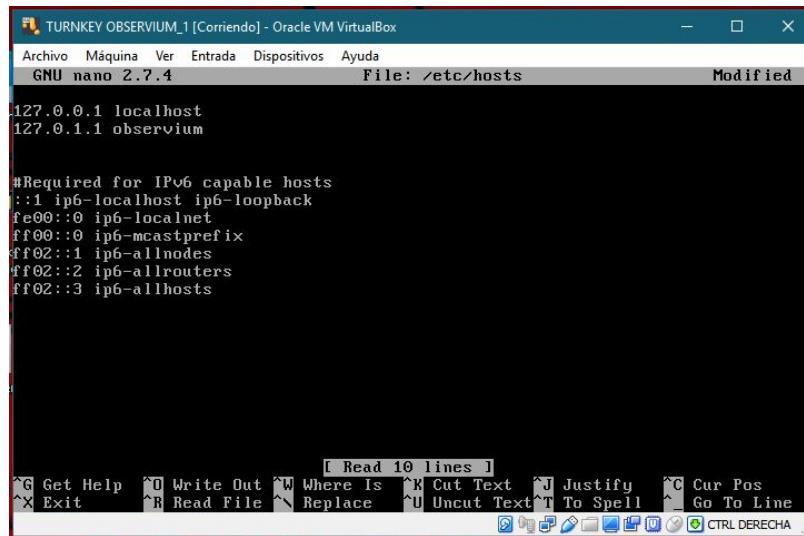


Figura 2.24: Modificación del archivo de agentes en Observium.

2. Obtenemos la dirección IP del agente de Linux.

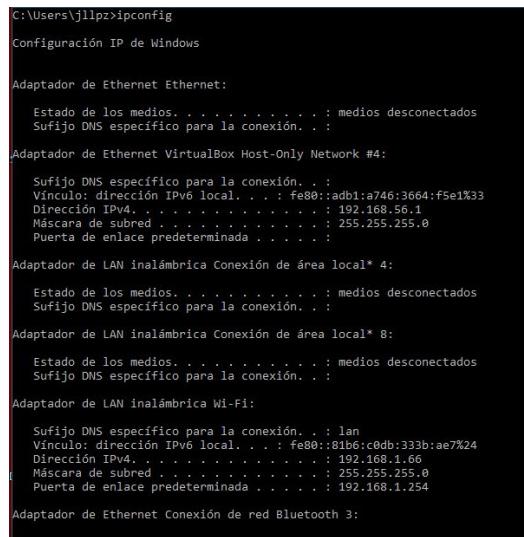
```
osboxes@osboxes:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 192.168.1.68 netmask 255.255.255.0 broadcast 192.168.1.255
        inet6 fe80::6403:4f64:bd64:23cd prefixlen 64 scopeid 0x20<link>
          ether 08:00:27:17:82:39 txqueuelen 1000 (Ethernet)
            RX packets 3307 bytes 1152510 (1.1 MB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 1222 bytes 121111 (121.1 KB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
        inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
          loop txqueuelen 1000 (Local Loopback)
            RX packets 747 bytes 63555 (63.5 KB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 747 bytes 63555 (63.5 KB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

osboxes@osboxes:~$
```

Figura 2.25: Obtención de la IP del agente de Linux.

3. Obtenemos la dirección IP del agente de Windows.



```
C:\Users\jlpz\ipconfig
Configuración IP de Windows

Adaptador de Ethernet Ethernet:
  Estado de los medios. . . . . : medios desconectados
  Sufijo DNS específico para la conexión. . .

Adaptador de Ethernet VirtualBox Host-Only Network #4:
  Sufijo DNS específico para la conexión. . .
  Vínculo: dirección IPv6 local. . . : fe80::adb1:a746:3664:f5e1%33
  Dirección IPv4. . . . . : 192.168.56.1
  Máscara de subred . . . . . : 255.255.255.0
  Puerta de enlace predeterminada . . .

Adaptador de LAN inalámbrica Conexión de área local* 4:
  Estado de los medios. . . . . : medios desconectados
  Sufijo DNS específico para la conexión. . .

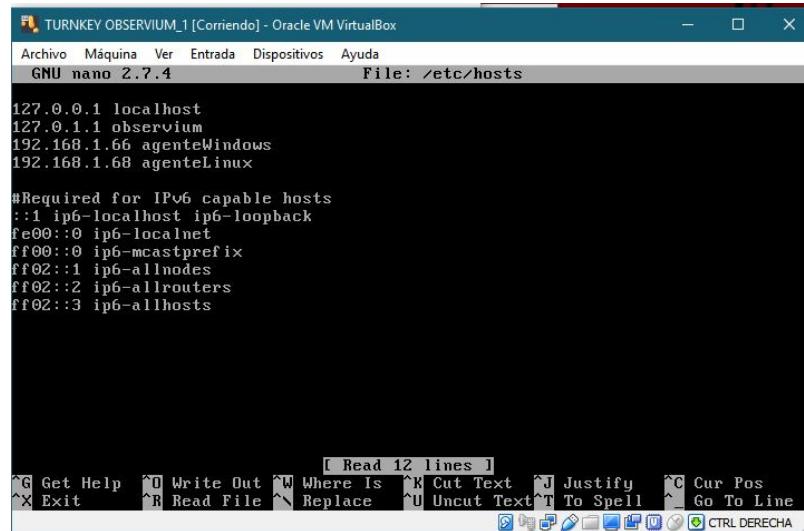
Adaptador de LAN inalámbrica Conexión de área local* 8:
  Estado de los medios. . . . . : medios desconectados
  Sufijo DNS específico para la conexión. . .

Adaptador de LAN inalámbrica Wi-Fi:
  Sufijo DNS específico para la conexión. . . : lan
  Vínculo: dirección IPv6 local. . . : fe80::81b6:c0d8:333b:ae7%24
  Dirección IPv4. . . . . : 192.168.1.66
  Máscara de subred . . . . . : 255.255.255.0
  Puerta de enlace predeterminada . . . . : 192.168.1.254

Adaptador de Ethernet Conexión de red Bluetooth 3:
```

Figura 2.26: Obtención de la IP del agente de Windows.

4. Agregamos la dirección y nombre del agente en el archivo de configuración *hosts* de Observium.



```
TURNKEY OBSERVIUM_1 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
GNU nano 2.7.4          File: /etc/hosts
127.0.0.1 localhost
127.0.1.1 observium
192.168.1.66 agenteWindows
192.168.1.68 agenteLinux

#Required for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts

[ Read 12 lines ]
^G Get Help ^O Write Out ^W Where Is ^X Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^P Replace ^U Uncut Text ^T To Spell ^_ Go To Line
CTRL DERECHA .
```

Figura 2.27: Agregación de la dirección y nombre del agente.

5. Verificamos que se agregaron que exista comunicación entre el gestor y los agentes mediante un ping.

```
root@observium ~# ping agenteWindows
PING agenteWindows (192.168.1.66) 56(84) bytes of data.
64 bytes from agenteWindows (192.168.1.66): icmp_seq=1 ttl=128 time=0.429 ms
64 bytes from agenteWindows (192.168.1.66): icmp_seq=2 ttl=128 time=0.505 ms
64 bytes from agenteWindows (192.168.1.66): icmp_seq=3 ttl=128 time=0.784 ms
64 bytes from agenteWindows (192.168.1.66): icmp_seq=4 ttl=128 time=0.627 ms
^X64 bytes from agenteWindows (192.168.1.66): icmp_seq=5 ttl=128 time=0.513 ms
64 bytes from agenteWindows (192.168.1.66): icmp_seq=6 ttl=128 time=0.510 ms
^Z
[11]+ Stopped                  ping agenteWindows
root@observium ~# ping agenteLinux
PING agenteLinux (192.168.1.68) 56(84) bytes of data.
64 bytes from agenteLinux (192.168.1.68): icmp_seq=1 ttl=64 time=0.501 ms
64 bytes from agenteLinux (192.168.1.68): icmp_seq=2 ttl=64 time=0.726 ms
64 bytes from agenteLinux (192.168.1.68): icmp_seq=3 ttl=64 time=0.492 ms
64 bytes from agenteLinux (192.168.1.68): icmp_seq=4 ttl=64 time=0.740 ms
^Z
[12]+ Stopped                  ping agenteLinux
root@observium ~# _
```

Figura 2.28: Probamos haciendo un ping a cada agente.

6. Ingresamos a Observium mediante el navegador con la dirección, en este caso de: *192.168.1.67*.  
 Ingresamos los datos para LogIn como administrador:

- Usuario: **admin**
- Password: **JllpzrMr<7**

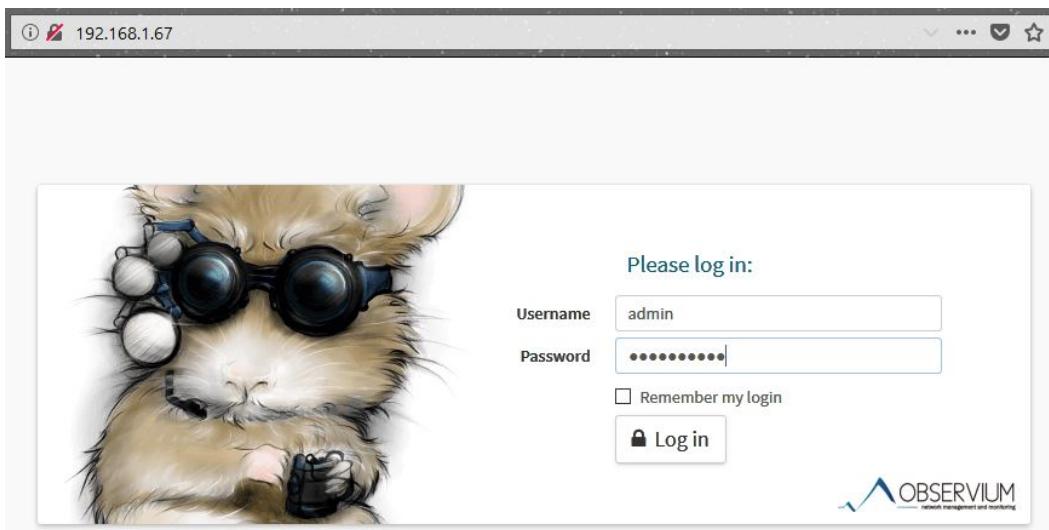


Figura 2.29: Login como Admin en la herramienta de Observium web.

7. Agregamos al agente Windows desde la interfaz de Observium, insertando la IP del agente, versión de SNMP y comunidad.

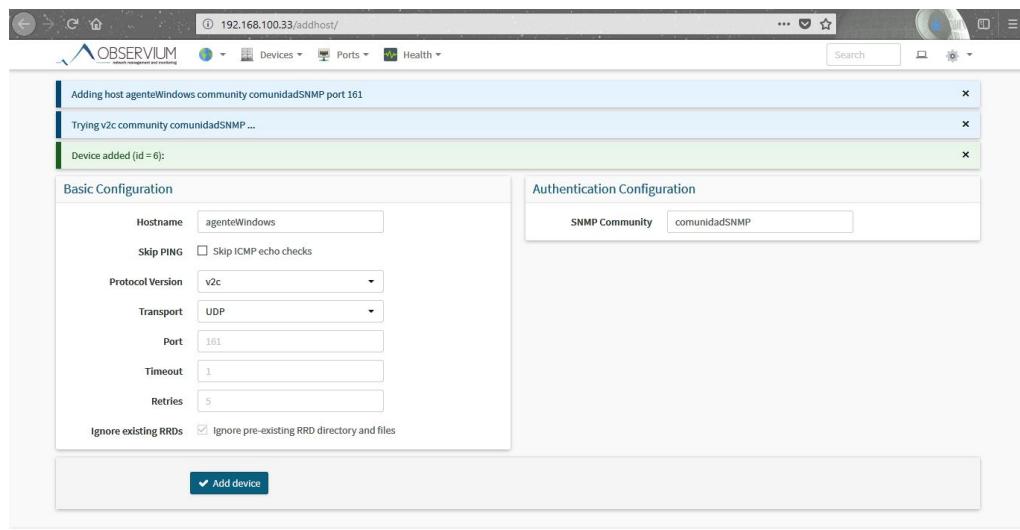


Figura 2.30: Agregación del agente Windows desde la plataforma web de Observium.

8. Agregamos al agente Linux agente desde la interfaz de Observium, insertando la IP del agente, versión de SNMP y comunidad.

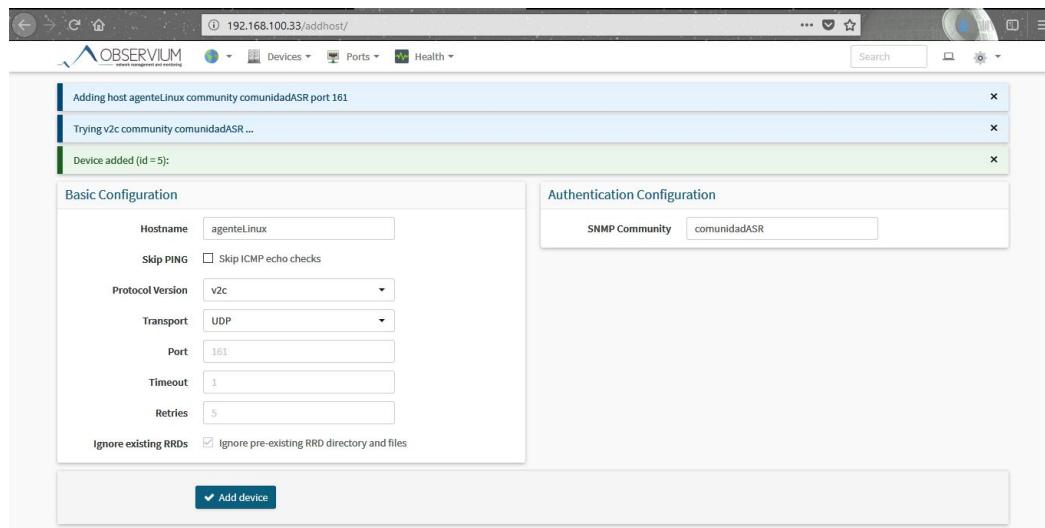


Figura 2.31: Agregación del agente Linux desde la plataforma web de Observium.

## 2.2. Resultados

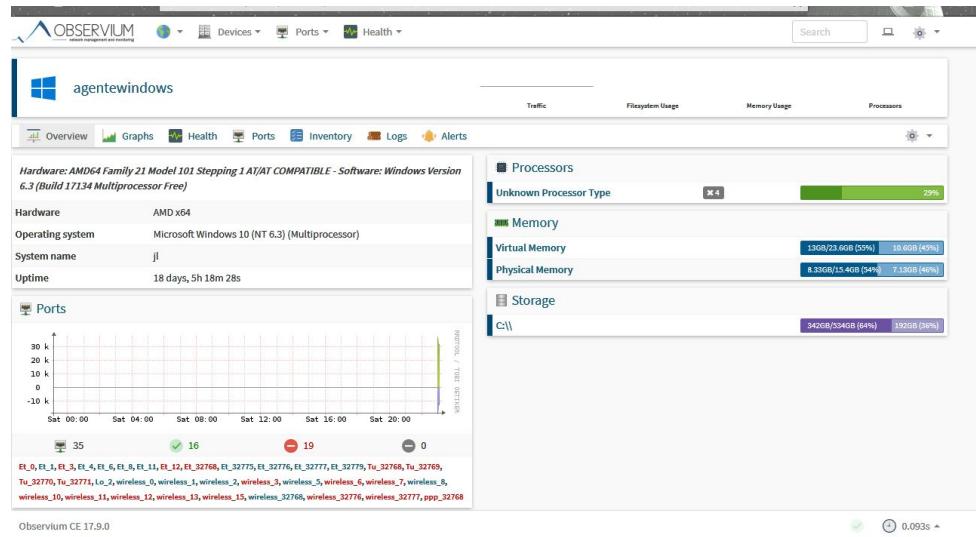


Figura 2.32: Resultados del agente Windows

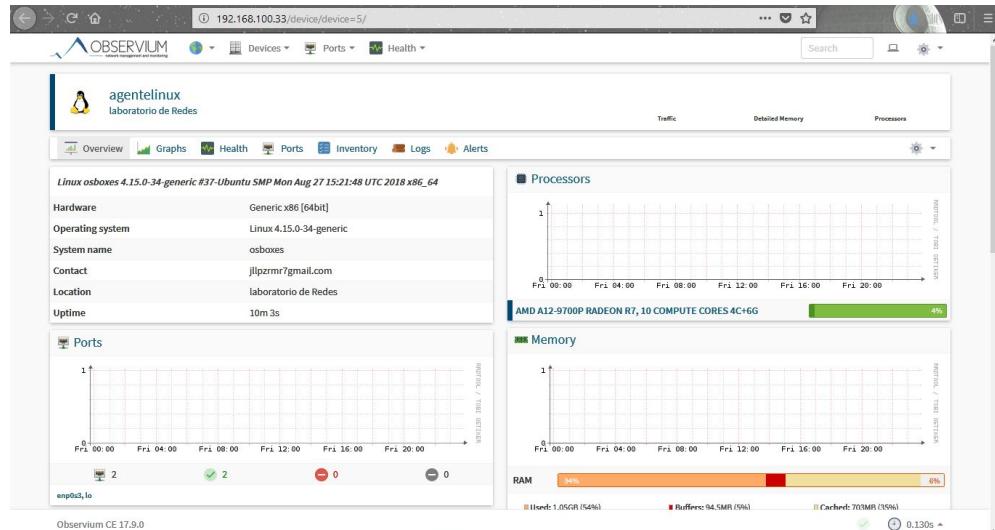


Figura 2.33: Resultados del agente Linux

## 2.3. Implementación de un modelo (versión 1) de administración de red de SNMP

Para el desarrollo de nuestra aplicación, usamos python con el framework Django y Celery. Así como rrdtool, pySNMP para la adquisición de datos de la MIB de cada agente.

Toda la aplicación se basa en Django, ya que este framework nos prevee de un desarrollo MvT, ademas de que nos permite el uso de la herramienta rrttool y pySNMP en python.

El objetivo principal de esta aplicación es monitorizar ciertos puntos de la MIB de un agente y esto debe ser en tiempo real, Celery nos permite realizar procesos asincronos sin necesidad de crear un multiThreading, es por esto que decidimos hacer uso de Celery.

A continuación se muestra como se compone el entorno de nuestra aplicación.

### 1) Inicio

El inicio de nuestra aplicación muestra un resumen general de todos los agentes a monitorizar. Este resumen contiene:

- 1. Número de agentes agregados
- 2. Estatus de conexión de cada dispositivo
- 3. SO de cada agente

Hostname	Versión SNMP	Puerto	Comunidad	OS	Estado	Opción
192.168.1.70	v2c	400	comunidadAmistades	Windows	Up	Monitorear
192.168.1.74	v2c	8000	comunidadAmistades	Linux	Up	Monitorear
localhost	v2c	800	comunidadAmistades	Linux	Up	Monitorear

Inicio de nuestra aplicación

## 2) Agregar agente

Los agentes que se agreguen al sistema son guardados en una base de datos, por lo cual el sistema es capaz de guardar múltiples agentes.

Para agregar un agente se debe indicar:

- 1. Hostname
- 2. Versión SNMP
- 3. Puerto SNMP
- 4. Comunidad

The screenshot shows a Mozilla Firefox browser window displaying the 'Añadir agente' (Add agent) page in the Django Admin interface. The URL in the address bar is `localhost:8000/admin/GestionAgentes/agente/add/`. The page title is 'Añadir agente | Sitio de administración de Django - Mozilla Firefox'. The main content area is titled 'Añadir agente' and contains four form fields:

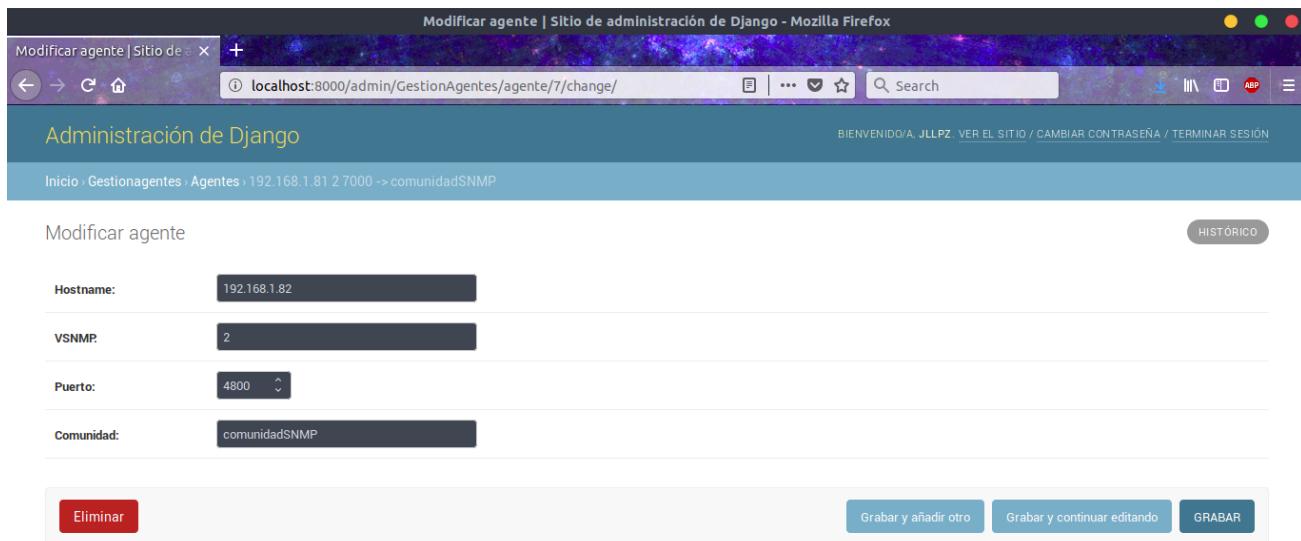
- Hostname: 192.168.1.81
- VSNMP: 2
- Puerto: 7000
- Comunidad: comunidadSNMP

At the bottom right of the form are three buttons: 'Grabar y añadir otro' (Save and add another), 'Grabar y continuar editando' (Save and continue editing), and a large blue button labeled 'GRABAR' (Save).

Agregar agentes

### 3) Eliminar agente

El sistema puede modificar la información de algún agente, ademas de eliminar.



The screenshot shows a Mozilla Firefox browser window with the title "Modificar agente | Sitio de administración de Django - Mozilla Firefox". The address bar shows "localhost:8000/admin/GestionAgentes/agente/7/change/". The main content area is titled "Administración de Django" and shows the "Modificar agente" form. The form has four fields: "Hostname" with value "192.168.1.82", "VSNMP" with value "2", "Puerto" with value "4800", and "Comunidad" with value "comunidadSNMP". Below the form are three buttons: a red "Eliminar" button, a blue "Grabar y añadir otro" button, and a blue "GRABAR" button.

### Eliminar y modificar agente

### 4) Estado de Dispositivo

El sistema indica la información principal del agente:

- nombre del host
- IP
- versión y logo del sistema operativo
- número de interfaces de red
- tiempo de actividad desde el último reinicio
- ubicación física
- información de contacto del administrador

También es posible visualizar el comportamiento del dispositivo mediante gráficas.

The screenshot shows a web browser window with the URL `localhost:8000/192.168.1.34/comunidadSNMP`. The page displays information about a network device named 'Agente JL'. On the left, there's a sidebar with user names ('López Romero Joel' and 'López Ayala Eric A.') and navigation links ('Inicio' and 'OPCIONES'). The main content area shows device details: Host: 192.168.1.34, Comunidad: comunidadSNMP, SO: Windows versión 6.3, No. de interfaces: 3, Tiempo activo: 12 days, Ubicación: Laboratorio Redes, Contacto: jllpzr7@gmail.com, and Estado: UP. The 'Estado' field is highlighted with a green bar.

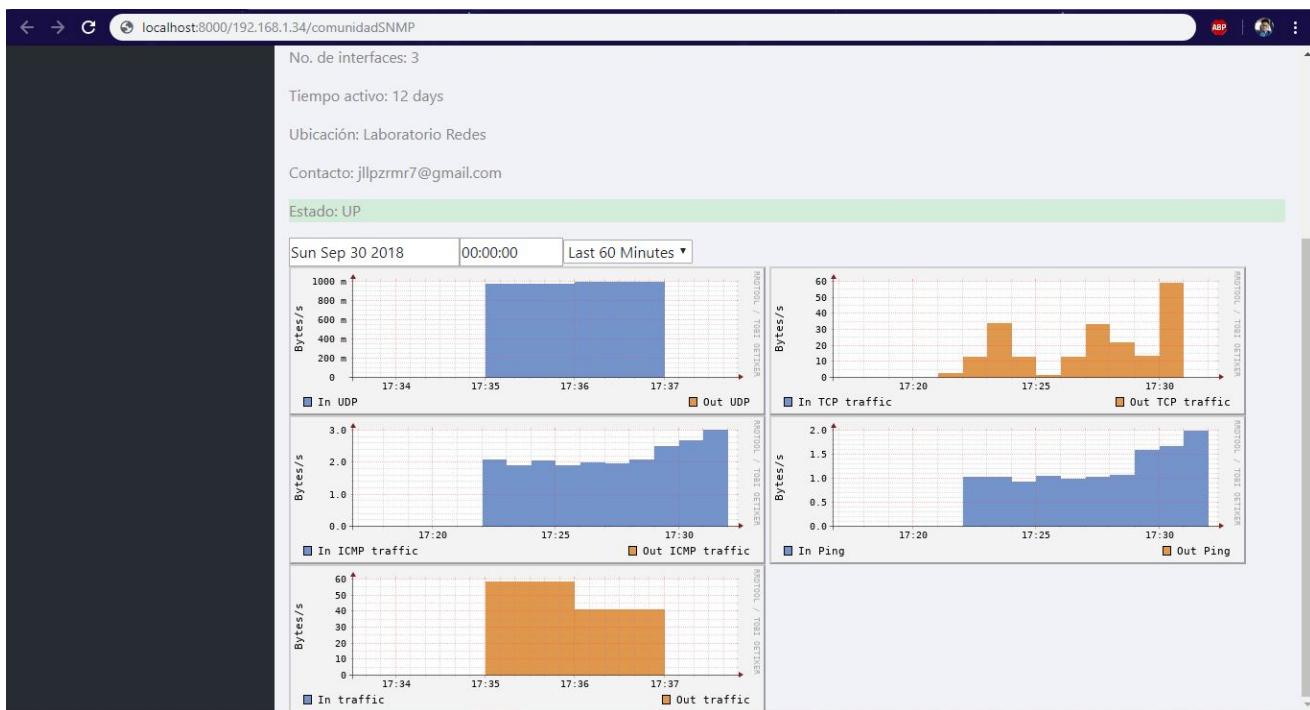
Estado del dispositivo

#### 4.1) Gráficas de dispositivo

El sistema muestra graficas generadas por rrdtool.

Elegimos los siguientes objetos de la MIB para describir su comportamiento:

- Tráfico en la interfaz de red eth0 de entrada: 1.3.6.1.2.1.2.2.1.10.1 y de salida 1.3.6.1.2.1.2.2.1.16.1
- Segmentos TCP de entrada: 1.3.6.1.2.1.6.10.0 y de salida: 1.3.6.1.2.1.6.11.0
- Datagramas UDP de entrada: 1.3.6.1.2.1.7.1.0' y de salida: 1.3.6.1.2.1.7.4.0'
- Segmentos ICMP de entrada: 1.3.6.1.2.1.5.1.0 y de salida: 1.3.6.1.2.1.5.14.0
- Lectura de Ping de entrada: 1.3.6.1.2.1.5.9.0 y ping de salida: 1.3.6.1.2.1.5.20.0



Eliminar y modificar agente

## 2.4. Cuestionario

### 2.4.1. Vigilancia y control de los agentes de gestión

#### Operaciones SNMP

Se generó los comandos SNMP correspondientes para contestar las siguientes preguntas, se hizo uso de 2 agentes:

- Agente Linux: *Ip: 192.168.1.81*
- Agente Windows: *Ip: 192.168.1.82*

Device / Location	Operating System / Hardware Platform	Uptime / sysName
 192.168.1.81 laboratorio de Redes	 5 Linux 4.15.0-34-generic Generic x86 [64bit]	5h 2m 30s headhunter
 192.168.1.82	 12 Microsoft Windows 7 SP1 (NT 6.1) (Multiprocessor) Generic x86	4h 23m eric-pc

Figura 2.34: Agentes utilizados.

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

Consulta Linux:

```
ESCOM@headhunter:/etc/snmp$ snmpget -v 2c -c comunidadSNMP 192.168.1.81 1.3.6.1.2.1.25.1.1.0
iso.3.6.1.2.1.25.1.1.0 = Timeticks: (259690) 0:43:16.90
```

Figura 2.35: Lectura de objeto sysUp Linux.

**Interpretación:** El último reinicio del equipo fue el día 29 de Septiembre a las 11 horas con 17 minutos.

Consulta Windows:

```
ESCOM@headhunter:/etc/snmp$ snmpget -v 2c -c comunidadSNMP 192.168.1.82 1.3.6.1.2.1.25.1.1.0
iso.3.6.1.2.1.25.1.1.0 = Timeticks: (26159) 0:04:21.59
```

Figura 2.36: Lectura de objeto sysUp Windows.

**Interpretación:** El último reinicio del equipo fue el día 29 de Septiembre a las 11 horas con 56 minutos.

2. ¿Cuántas interfaces Ethernet tiene el agente?

Especificación en el RFC 1213 - IEFT:

```
ifType OBJECT-TYPE
    SYNTAX INTEGER {
        other(1),
        regular1822(2),
        hdh1822(3),
        ddn-x25(4),
        rfc877-x25(5),
        ethernet-csmacd(6),
        iso88023-csmacd(7),
        ...others...
    }
    -- none of the following
```

Figura 2.37: Especificación del objeto interfaces- ifType.

Consulta Linux:

```
ESCOM@headhunter:/etc/snmp$ snmpwalk -v 2c -c comunidadSNMP 192.168.1.81 1.3.6.1.2.1.2.2.1.3
iso.3.6.1.2.1.2.2.1.3.1 = INTEGER: 24
iso.3.6.1.2.1.2.2.1.3.2 = INTEGER: 6
iso.3.6.1.2.1.2.2.1.3.3 = INTEGER: 6
iso.3.6.1.2.1.2.2.1.3.4 = INTEGER: 6
iso.3.6.1.2.1.2.2.1.3.5 = INTEGER: 6
```

Figura 2.38: Lectura de objeto ifType Linux.

**Interpretación:** Dispositivos ethernet.

- 1.3.6.1.2.1.2.2.1.3.2
- 1.3.6.1.2.1.2.2.1.3.3
- 1.3.6.1.2.1.2.2.1.3.4
- 1.3.6.1.2.1.2.2.1.3.5

**Resultado: 4**

Consulta Windows:

```
ESCOM@headhunter:/etc/snmp$ snmpwalk -v 2c -c comunidadSNMP 192.168.1.82 1.3.6.1.2.1.2.2.1.3
iso.3.6.1.2.1.2.2.1.3.1 = INTEGER: 24
iso.3.6.1.2.1.2.2.1.3.2 = INTEGER: 131
iso.3.6.1.2.1.2.2.1.3.3 = INTEGER: 131
iso.3.6.1.2.1.2.2.1.3.4 = INTEGER: 131
iso.3.6.1.2.1.2.2.1.3.5 = INTEGER: 23
iso.3.6.1.2.1.2.2.1.3.6 = INTEGER: 6
iso.3.6.1.2.1.2.2.1.3.7 = INTEGER: 6
iso.3.6.1.2.1.2.2.1.3.8 = INTEGER: 6
iso.3.6.1.2.1.2.2.1.3.9 = INTEGER: 23
iso.3.6.1.2.1.2.2.1.3.10 = INTEGER: 131
iso.3.6.1.2.1.2.2.1.3.11 = INTEGER: 6
iso.3.6.1.2.1.2.2.1.3.12 = INTEGER: 131
iso.3.6.1.2.1.2.2.1.3.13 = INTEGER: 131
iso.3.6.1.2.1.2.2.1.3.14 = INTEGER: 131
iso.3.6.1.2.1.2.2.1.3.15 = INTEGER: 6
iso.3.6.1.2.1.2.2.1.3.16 = INTEGER: 6
iso.3.6.1.2.1.2.2.1.3.17 = INTEGER: 6
iso.3.6.1.2.1.2.2.1.3.18 = INTEGER: 6
iso.3.6.1.2.1.2.2.1.3.19 = INTEGER: 6
```

Figura 2.39: Lectura de objeto ifType Windows.

**Interpretación:** Dispositivos ethernet.

- 1.3.6.1.2.1.2.2.1.3.6
- 1.3.6.1.2.1.2.2.1.3.7
- 1.3.6.1.2.1.2.2.1.3.8
- 1.3.6.1.2.1.2.2.1.3.11
- 1.3.6.1.2.1.2.2.1.3.15
- 1.3.6.1.2.1.2.2.1.3.16
- 1.3.6.1.2.1.2.2.1.3.17
- 1.3.6.1.2.1.2.2.1.3.18
- 1.3.6.1.2.1.2.2.1.3.19

**Resultado: 9**

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

Especificación en el RFC 1213 - IEFT:

```
ifSpeed OBJECT-TYPE
    SYNTAX  Gauge
    ACCESS  read-only
    STATUS   mandatory
    DESCRIPTION
        "An estimate of the interface's current bandwidth
         in bits per second. For interfaces which do not
         vary in bandwidth or for those where no accurate
         estimation can be made, this object should contain
         the nominal bandwidth."
 ::= { ifEntry 5 }
```

Figura 2.40: Especificación del objeto interfaces- ifSpeed.

Consulta Linux:

```
ESCOM@headhunter:/etc/snmp$ snmpwalk -v 2c -c comunidadSNMP 192.168.1.81 1.3.6.1.2.1.2.2.1.5
iso.3.6.1.2.1.2.2.1.5.1 = Gauge32: 10000000
iso.3.6.1.2.1.2.2.1.5.2 = Gauge32: 100000000
iso.3.6.1.2.1.2.2.1.5.3 = Gauge32: 0
iso.3.6.1.2.1.2.2.1.5.4 = Gauge32: 0
iso.3.6.1.2.1.2.2.1.5.5 = Gauge32: 10000000
```

Figura 2.41: Lectura de objeto ifSpeed Linux.

**Interpretación:** Velocidad de los dispositivos ethernet.

- 1.3.6.1.2.1.2.2.1.3.2 - 100000000 b/s ->100MBPS
- 1.3.6.1.2.1.2.2.1.3.3 - 0 b/s ->0MBPS
- 1.3.6.1.2.1.2.2.1.3.4 - 0 b/s ->0MBPS
- 1.3.6.1.2.1.2.2.1.3.5 - 10000000 b/s ->10MBPS

Consulta Windows:

```
ESCOM@headhunter:/etc/snmp$ snmpwalk -v 2c -c comunidadSNMP 192.168.1.82 1.3.6.1.2.1.2.2.1.5
iso.3.6.1.2.1.2.2.1.5.1 = Gauge32: 1073741824
iso.3.6.1.2.1.2.2.1.5.2 = Gauge32: 1073741824
iso.3.6.1.2.1.2.2.1.5.3 = Gauge32: 1073741824
iso.3.6.1.2.1.2.2.1.5.4 = Gauge32: 1073741824
iso.3.6.1.2.1.2.2.1.5.5 = Gauge32: 1073741824
iso.3.6.1.2.1.2.2.1.5.6 = Gauge32: 1073741824
iso.3.6.1.2.1.2.2.1.5.7 = Gauge32: 1073741824
iso.3.6.1.2.1.2.2.1.5.8 = Gauge32: 1073741824
iso.3.6.1.2.1.2.2.1.5.9 = Gauge32: 0
iso.3.6.1.2.1.2.2.1.5.10 = Gauge32: 0
iso.3.6.1.2.1.2.2.1.5.11 = Gauge32: 10000000000
iso.3.6.1.2.1.2.2.1.5.12 = Gauge32: 300000000
iso.3.6.1.2.1.2.2.1.5.13 = Gauge32: 100000
iso.3.6.1.2.1.2.2.1.5.14 = Gauge32: 100000
iso.3.6.1.2.1.2.2.1.5.15 = Gauge32: 10000000000
iso.3.6.1.2.1.2.2.1.5.16 = Gauge32: 10000000000
iso.3.6.1.2.1.2.2.1.5.17 = Gauge32: 1073741824
iso.3.6.1.2.1.2.2.1.5.18 = Gauge32: 1073741824
iso.3.6.1.2.1.2.2.1.5.19 = Gauge32: 1073741824
```

Figura 2.42: Lectura de objeto ifSpeed Windows.

**Interpretación:** Velocidad de los dispositivos ethernet.

- 1.3.6.1.2.1.2.2.1.3.6 - 1073741824 b/s ->1.07GBPS
- 1.3.6.1.2.1.2.2.1.3.7 - 1073741824 b/s ->1.07GBPS
- 1.3.6.1.2.1.2.2.1.3.8 - 1073741824 b/s ->1.07GBPS
- 1.3.6.1.2.1.2.2.1.3.11 - 1000000000 b/s ->1GBPS
- 1.3.6.1.2.1.2.2.1.3.15 - 1000000000 b/s ->1GBPS
- 1.3.6.1.2.1.2.2.1.3.16 - 1000000000 b/s ->1GBPS
- 1.3.6.1.2.1.2.2.1.3.17 - 1073741824 b/s ->1.07GBPS
- 1.3.6.1.2.1.2.2.1.3.18 - 1073741824 b/s ->1.07GBPS
- 1.3.6.1.2.1.2.2.1.3.19 - 1073741824 b/s ->1.07GBPS

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

Especificación en el RFC 1213 - IEFT:

```
ifInOctets OBJECT-TYPE
  SYNTAX  Counter
  ACCESS  read-only

  ifInOctets  Group
  [Page 20]
  MIB-II          March 1991

  STATUS mandatory
  DESCRIPTION
    "The total number of octets received on the
     interface, including framing characters."
  ::= { ifEntry 10 }
```

Figura 2.43: Especificación del objeto interfaces - ifInOctets.

Consulta Linux:

```
ESCOM@headhunter:/etc/snmp$ snmpwalk -v 2c -c comunidadSNMP 192.168.1.81 1.3.6.1.2.1.2.2.1.10
iso.3.6.1.2.1.2.2.1.10.1 = Counter32: 351517
iso.3.6.1.2.1.2.2.1.10.2 = Counter32: 434071868
iso.3.6.1.2.1.2.2.1.10.3 = Counter32: 1377351
iso.3.6.1.2.1.2.2.1.10.4 = Counter32: 0
iso.3.6.1.2.1.2.2.1.10.5 = Counter32: 0
```

Figura 2.44: Lectura de objeto ifInOctets Linux.

**Interpretación:** Número de octetos que ha recibido la interfaz.

- 1.3.6.1.2.1.2.2.1.10.2 - 4340771868 octetos

Consulta Windows:

```
ESCOM@headhunter:/etc/snmp$ snmpwalk -v 2c -c comunidadSNMP 192.168.1.82 1.3.6.1.2.1.2.2.1.10
iso.3.6.1.2.1.2.2.1.10.1 = Counter32: 0
iso.3.6.1.2.1.2.2.1.10.2 = Counter32: 0
iso.3.6.1.2.1.2.2.1.10.3 = Counter32: 0
iso.3.6.1.2.1.2.2.1.10.4 = Counter32: 0
iso.3.6.1.2.1.2.2.1.10.5 = Counter32: 0
iso.3.6.1.2.1.2.2.1.10.6 = Counter32: 0
iso.3.6.1.2.1.2.2.1.10.7 = Counter32: 0
iso.3.6.1.2.1.2.2.1.10.8 = Counter32: 0
iso.3.6.1.2.1.2.2.1.10.9 = Counter32: 0
iso.3.6.1.2.1.2.2.1.10.10 = Counter32: 0
iso.3.6.1.2.1.2.2.1.10.11 = Counter32: 261541681
iso.3.6.1.2.1.2.2.1.10.12 = Counter32: 0
iso.3.6.1.2.1.2.2.1.10.13 = Counter32: 0
iso.3.6.1.2.1.2.2.1.10.14 = Counter32: 0
iso.3.6.1.2.1.2.2.1.10.15 = Counter32: 261541681
iso.3.6.1.2.1.2.2.1.10.16 = Counter32: 261541681
iso.3.6.1.2.1.2.2.1.10.17 = Counter32: 0
iso.3.6.1.2.1.2.2.1.10.18 = Counter32: 0
iso.3.6.1.2.1.2.2.1.10.19 = Counter32: 0
```

Figura 2.45: Lectura de objeto ifInOctets Windows.

**Interpretación:** Número de octetos que ha recibido la interfaz.

- 1.3.6.1.2.1.2.2.1.10.11 - 251541681 octetos
- 1.3.6.1.2.1.2.2.1.10.15 - 251541681 octetos
- 1.3.6.1.2.1.2.2.1.10.16 - 251541681 octetos

5. Indica el número de octetos de la interfaz que ha recibido el mayor número de octetos - **La respuesta se encuentra en la pregunta 4.**

6. ¿Cuál es la MAC de esa interfaz?

Especificación en el RFC 1213 - IEFT:

```

ifPhysAddress OBJECT-TYPE
  SYNTAX  PhysAddress
  ACCESS  read-only
  STATUS   mandatory
  DESCRIPTION
    "The interface's address at the protocol layer
     immediately 'below' the network layer in the
     protocol stack. For interfaces which do not have
     such an address (e.g., a serial line), this object
     should contain an octet string of zero length."

```

ing Group	[Page 19]
MIB-II	March 1991

Figura 2.46: Especificación del objeto interfaces - ifPhysAddress.

Consulta Linux:

```
ESCOM@headhunter:/etc/snmp$ snmpget -v 2c -c comunidadSNMP 192.168.1.81 1.3.6.1.2.1.2.2.1.6.2iso
.3.6.1.2.1.2.2.1.6.2 = Hex-STRING: 50 7B 9D 4B 56 A7
```

Figura 2.47: Lectura de objeto ifPhysAddress Linux.

**Interpretación:** Dirección física de la interfaz.

- 1.3.6.1.2.1.2.2.1.10.2 - 50 7B 9D 4B 56 A7

Consulta Windows:

```
ESCOM@headhunter:/etc/snmp$ snmpget -v 2c -c comunidadSNMP 192.168.1.82 1.3.6.1.2.1.2.2.1.6.11
iso.3.6.1.2.1.2.2.1.6.11 = Hex-STRING: 08 00 27 1B BE 48
ESCOM@headhunter:/etc/snmp$ snmpget -v 2c -c comunidadSNMP 192.168.1.82 1.3.6.1.2.1.2.2.1.6.15
iso.3.6.1.2.1.2.2.1.6.15 = Hex-STRING: 08 00 27 1B BE 48
ESCOM@headhunter:/etc/snmp$ snmpget -v 2c -c comunidadSNMP 192.168.1.82 1.3.6.1.2.1.2.2.1.6.16
iso.3.6.1.2.1.2.2.1.6.16 = Hex-STRING: 08 00 27 1B BE 48
```

Figura 2.48: Lectura de objeto ifPhysAddress Windows.

**Interpretación:** Dirección física de la interfaz.

- 1.3.6.1.2.1.2.2.1.10.11 - 08 00 27 1B BE 48
- 1.3.6.1.2.1.2.2.1.10.15 - 08 00 27 1B BE 48
- 1.3.6.1.2.1.2.2.1.10.16 - 08 00 27 1B BE 48

7. ¿Cuántos mensajes ICMP ha recibido el agente?

Especificación en el RFC 1213 - IEFT:

```
icmpInMsgs OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of ICMP messages which the
         entity received. Note that this counter includes
         all those counted by icmpInErrors."
```

Figura 2.49: Especificación del objeto ICMP - inMsgs.

Consulta Linux:

```
ESCOM@headhunter:/etc/snmp$ snmpget -v 2c -c comunidadSNMP 192.168.1.81 1.3.6.1.2.1.5.1.0
iso.3.6.1.2.1.5.1.0 = Counter32: 113
```

Figura 2.50: Lectura de objeto icmpInMsgs Linux.

**Interpretación:** El agente ha recibido 113 mensajes ICMP.

Consulta Windows:

```
ESCOM@headhunter:/etc/snmp$ snmpget -v 2c -c comunidadSNMP 192.168.1.82 1.3.6.1.2.1.5.1.0
iso.3.6.1.2.1.5.1.0 = Counter32: 2
```

Figura 2.51: Lectura de objeto icmpInMsgs Windows.

**Interpretación:** El agente ha recibido 2 mensajes ICMP.

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

Especificación en el RFC 1213 - IEFT:

```
ipRouteDest OBJECT-TYPE
  SYNTAX  IpAddress
  ACCESS  read-write
  STATUS   mandatory
  DESCRIPTION
    "The destination IP address of this route. An
     entry with a value of 0.0.0.0 is considered a
     default route. Multiple routes to a single
     destination can appear in the table, but access to
     such multiple entries is dependent on the table-
     access mechanisms defined by the network
     management protocol in use."
```

Figura 2.52: Especificación del objeto IP - routeDest.

Consulta Linux:

```
ESCOM@headhunter:/etc/snmp$ snmpwalk -v 2c -c comunidadSNMP 192.168.1.81 1.3.6.1.2.1.4.21.1.1
iso.3.6.1.2.1.4.21.1.1.0.0.0.0 =IpAddress: 0.0.0.0
iso.3.6.1.2.1.4.21.1.1.169.254.0.0 =IpAddress: 169.254.0.0
iso.3.6.1.2.1.4.21.1.1.192.168.1.0 =IpAddress: 192.168.1.0
iso.3.6.1.2.1.4.21.1.1.192.168.123.0 =IpAddress: 192.168.123.0
```

Figura 2.53: Lectura de objeto ipRouteDest Linux.

**Interpretación:** El agente tiene 4 entradas en la tabla de enrutamiento.

Consulta Windows:

```
ESCOM@headhunter:/etc/snmp$ snmpwalk -v 2c -c comunidadSNMP 192.168.1.82 1.3.6.1.2.1.4.21.1.1
iso.3.6.1.2.1.4.21.1.1.0.0.0.0 =IpAddress: 0.0.0.0
iso.3.6.1.2.1.4.21.1.1.127.0.0.0 =IpAddress: 127.0.0.0
iso.3.6.1.2.1.4.21.1.1.127.0.0.1 =IpAddress: 127.0.0.1
iso.3.6.1.2.1.4.21.1.1.127.255.255.255 =IpAddress: 127.255.255.255
iso.3.6.1.2.1.4.21.1.1.192.168.1.0 =IpAddress: 192.168.1.0
iso.3.6.1.2.1.4.21.1.1.192.168.1.82 =IpAddress: 192.168.1.82
iso.3.6.1.2.1.4.21.1.1.192.168.1.255 =IpAddress: 192.168.1.255
iso.3.6.1.2.1.4.21.1.1.224.0.0.0 =IpAddress: 224.0.0.0
iso.3.6.1.2.1.4.21.1.1.255.255.255.255 =IpAddress: 255.255.255.255
```

Figura 2.54: Lectura de objeto ipRouteDest Windows.

**Interpretación:** El agente tiene 9 entradas en la tabla de enrutamiento.

9. ¿Cuántos datagramas UDP ha recibido el agente?

Especificación en el RFC 1213 - IEFT:

```
udpInDatagrams OBJECT-TYPE
    SYNTAX  Counter
    ACCESS  read-only
    STATUS   mandatory
    DESCRIPTION
        "The total number of UDP datagrams delivered to
         UDP users."
```

Figura 2.55: Especificación del objeto UDP - inDatagrams.

Consulta Linux:

```
ESCOM@headhunter:/etc/snmp$ snmpget -v 2c -c comunidadSNMP 192.168.1.81 1.3.6.1.2.1.7.1.0
iso.3.6.1.2.1.7.1.0 = Counter32: 18797
```

Figura 2.56: Lectura de objeto udpInDatagrams Linux.

**Interpretación:** El agente ha recibido 18797 datagramas UDP.

Consulta Windows:

```
ESCOM@headhunter:/etc/snmp$ snmpget -v 2c -c comunidadSNMP 192.168.1.82 1.3.6.1.2.1.7.1.0
iso.3.6.1.2.1.7.1.0 = Counter32: 2808
```

Figura 2.57: Lectura de objeto udpInDatagrams Windows.

**Interpretación:** El agente ha recibido 2808 datagramas UDP.

10. ¿El agente ha recibido mensajes TCP? ¿Cuántos?

Especificación en el RFC 1213 - IEFT:

```
tcpInSegs OBJECT-TYPE
  SYNTAX Counter
  ACCESS read-only
  STATUS mandatory
  DESCRIPTION
    "The total number of segments received, including
     those received in error. This count includes
     segments received on currently established
     connections."
```

Figura 2.58: Especificación del objeto TCP - inSegs.

Consulta Linux:

```
ESCOM@headhunter:/etc/snmp$ snmpget -v 2c -c comunidadSNMP 192.168.1.81 1.3.6.1.2.1.6.10.0
iso.3.6.1.2.1.6.10.0 = Counter32: 390293
```

Figura 2.59: Lectura de objeto tcpInSegs Linux.

**Interpretación:** El agente ha recibido 390293 segmentos TCP.

Consulta Windows:

```
ESCOM@headhunter:/etc/snmp$ snmpget -v 2c -c comunidadSNMP 192.168.1.82 1.3.6.1.2.1.6.10.0
iso.3.6.1.2.1.6.10.0 = Counter32: 427469
```

Figura 2.60: Lectura de objeto tcpInSegs Windows.

**Interpretación:** El agente ha recibido 427469 segmentos TCP.

11. ¿Cuántos mensajes EGP ha recibido el agente?

## Especificación en el RFC 1213 - IEFT:

```
egpInMsgs OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The number of EGP messages received without
         error."
```

Figura 2.61: Especificación del objeto EGP - inMsgs.

Consulta Linux/Windows:

```
ESCOM@headhunter:/etc/snmp$ snmpget -v 2c -c comunidadSNMP 192.168.1.81 1.3.6.1.2.1.8.1.0  
iso.3.6.1.2.1.8.1.0 = No Such Object available on this agent at this OID  
ESCOM@headhunter:/etc/snmp$ snmpget -v 2c -c comunidadSNMP 192.168.1.82 1.3.6.1.2.1.8.1.0  
iso.3.6.1.2.1.8.1.0 = No Such Object available on this agent at this OID
```

Figura 2.62: Lectura de objeto egpInMsgs Linux/Windows.

Figura 2.63: Evidencia de que no existe el objeto eppInMsgs en la MIB del agente.

**Interpretación:** La variable no existe en la MIB del agente, debido a que no implementa el protocolo EGP.

12. Indica el Sistema Operativo que maneja el agente.

Especificación en el RFC 1213 - IEFT:

```
sysDescr OBJECT-TYPE
    SYNTAX  DisplayString (SIZE (0..255))
    ACCESS  read-only
    STATUS  mandatory

    ing Group          [Page 13]
    MIB-II           March 1991

    DESCRIPTION
        "A textual description of the entity. This value
        should include the full name and version
        identification of the system's hardware type,
        software operating-system, and networking
        software. It is mandatory that this only contain
        printable ASCII characters."
```

Figura 2.64: Especificación del objeto system - sysDescr.

Consulta Linux:

```
ESCOM@headhunter:/etc/snmp$ snmpget -v 2c -c comunidadSNMP 192.168.1.81 1.3.6.1.2.1.1.1.0
iso.3.6.1.2.1.1.1.0 = STRING: "Linux headhunter 4.15.0-34-generic #37-Ubuntu SMP Mon Aug 27 15:2
1:48 UTC 2018 x86_64"
```

Figura 2.65: Lectura de objeto sysDescr Linux.

**Interpretación:** Versión de Software - Linux 4.15.0-34-generic Ubuntu.

Consulta Windows:

```
ESCOM@headhunter:/etc/snmp$ snmpget -v 2c -c comunidadSNMP 192.168.1.82 1.3.6.1.2.1.1.1.0
iso.3.6.1.2.1.1.1.0 = STRING: "Hardware: x86 Family 22 Model 48 Stepping 1 AT/AT COMPATIBLE - So
ftware: Windows Version 6.1 (Build 7601 Multiprocessor Free)"
```

Figura 2.66: Lectura de objeto sysDescr Windows.

**Interpretación:** Versión de Software - Windows 6.1 - Windows 7.

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

Especificación en el RFC 1213 - IEFT:

```
sysContact OBJECT-TYPE
    SYNTAX  DisplayString (SIZE (0..255))
    ACCESS  read-write
    STATUS   mandatory
    DESCRIPTION
        "The textual identification of the contact person
         for this managed node, together with information
         on how to contact this person."
```

Figura 2.67: Especificación del objeto system - sysContact.

Consulta y Set al Agente:

```
ESCOM@headhunter:/etc/snmp$ snmpget -v 2c -c comunidadSNMP 192.168.1.82 1.3.6.1.2.1.1.4.0
iso.3.6.1.2.1.1.4.0 = STRING: "eralejandroayala@gmail.com"
ESCOM@headhunter:/etc/snmp$ snmpset -v 2c -c comunidadSNMP 192.168.1.82 1.3.6.1.2.1.1.4.0 s "over
rlord.lae@gmail.com"
iso.3.6.1.2.1.1.4.0 = STRING: "overlord.lae@gmail.com"
ESCOM@headhunter:/etc/snmp$ snmpget -v 2c -c comunidadSNMP 192.168.1.82 1.3.6.1.2.1.1.4.0
iso.3.6.1.2.1.1.4.0 = STRING: "overlord.lae@gmail.com"
```

Figura 2.68: Lectura y Escritura del objeto sysContact del Agente.

Interpretación:

- Se realizo la lectura del objeto sysContact del Agente - *Resultado: eralejandroayala@gmail.com*"
- Se realizo la escritura del objeto sysContact del Agente - *Resultado: overlord.lae@gmail.com*"
- Se realizo la lectura del objeto sysContact del Agente - *Resultado: overlord.lae@gmail.com*"

14. Dibuja la MIB del agente

## 2.5. Análisis de tráfico

Para el análisis del tráfico se monitorearon los paquetes SNMP entre el agente y el gestor, mediante el uso de la herramienta Wireshark, capturando los comandos básicos de SNMP.

### 2.5.1. Captura de paquetes SNMP formados por el comando snmpget

Consulta en consola usando comando **snmpget**

```
ESCOM@headhunter:~$ snmpget -v 2c -c comunidadSNMP 192.162.1.82 1.3.6.1.2.1.1.3.0
```

Figura 2.69: Captura del comando snmpget en consola.

Visualización del paquete SNMP en Wireshark

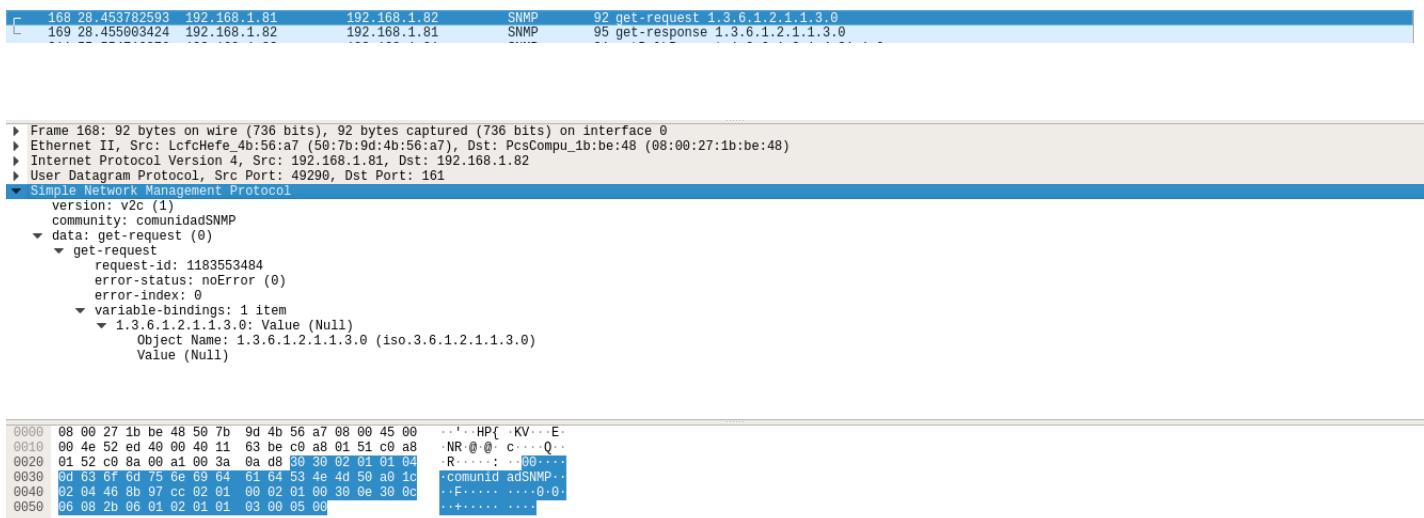


Figura 2.70: Capturas del comando snmpget en Wireshark.

## 2.5.2. Captura de paquetes SNMP formados por el comando snmpset

Consulta en consola usando comando **snmpset**

```
ESCOM@headhunter:~$ snmpset -v 2c -c comunidadSNMP 192.162.1.82 1.3.6.1.2.1.1.4.0 s "overlord.lae@gmail.com"
```

Figura 2.71: Captura del comando snmpset en consola.

Visualización del paquete SNMP en Wireshark

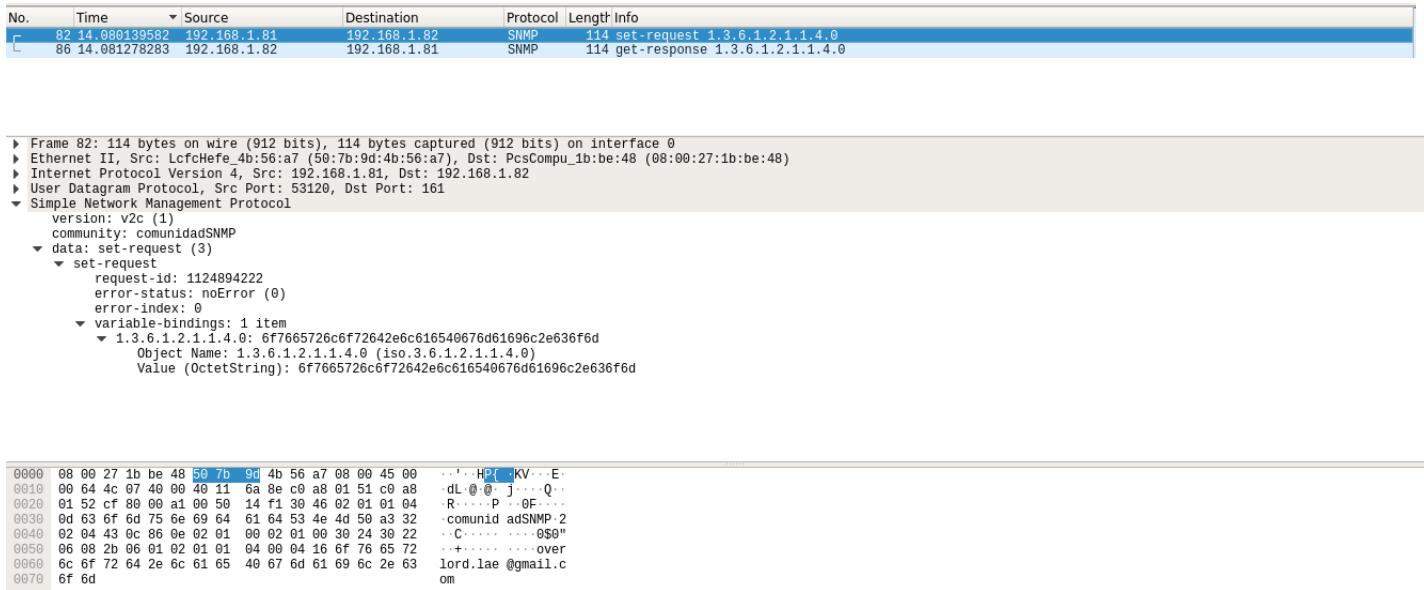


Figura 2.72: Capturas del comando snmpset en Wireshark.

### 2.5.3. Captura de paquetes SNMP formados por el comando snmpgetnext

Consulta en consola usando comando **snmpgetnext**

```
ESCOM@headhunter:~$ snmpgetnext -v 2c -c comunidadSNMP 192.168.1.82 1.3.6.1.2.1.1.3
```

Figura 2.73: Captura del comando snmpgetnext en consola.

Visualización del paquete SNMP en Wireshark

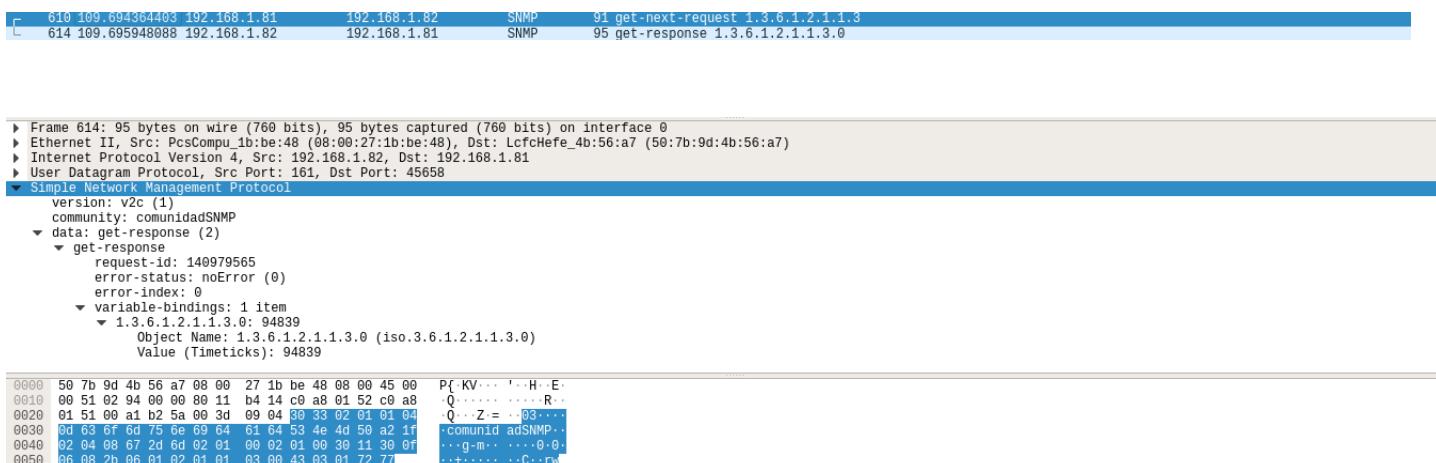


Figura 2.74: Capturas del comando snmpgetnext en Wireshark.

## 2.5.4. Captura de paquetes SNMP formados por el comando snmptable

Consulta en consola usando comando **snmptable**

```
ESCOM@headhunter:~$ snmptable -v 2c -c comunidadSNMP 192.162.1.82 1.3.6.1.2.1.4.21.1.1.192.16...
```

Figura 2.75: Captura del comando snmptable en consola.

Visualización del paquete SNMP en Wireshark

312 55.555529862 192.168.1.81	192.168.1.82	SNMP	306 get-response 1.3.6.1.2.1.4.21.1.1.0.0.0.0 1.3.6.1.2.1.4.21.1.1.169.254.0.0 1.3.6.1.2.1.4.21.1.1.192.16...
313 55.556215234 192.168.1.82	192.168.1.81	SNMP	97 getBulkRequest 1.3.6.1.2.1.4.21.1.3.169.254.0.0
314 55.556570920 192.168.1.81	192.168.1.82	SNMP	308 get-response 1.3.6.1.2.1.4.21.1.3.192.168.1.0 1.3.6.1.2.1.4.21.1.3.192.168.123.0 1.3.6.1.2.1.4.21.1.7...
315 55.556977813 192.168.1.82	192.168.1.81	SNMP	97 getBulkRequest 1.3.6.1.2.1.4.21.1.8.192.168.123.0
316 55.557199558 192.168.1.81	192.168.1.82	SNMP	306 get-response 1.3.6.1.2.1.4.21.1.9.0.0.0.0 1.3.6.1.2.1.4.21.1.9.169.254.0.0 1.3.6.1.2.1.4.21.1.9.192.16...
317 55.557565432 192.168.1.82	192.168.1.81	SNMP	97 getBulkRequest 1.3.6.1.2.1.4.21.1.13.169.254.0.0
318 55.558038896 192.168.1.81	192.168.1.82	SNMP	332 get-response 1.3.6.1.2.1.4.21.1.13.192.168.1.0 1.3.6.1.2.1.4.21.1.13.192.168.123.0 1.3.6.1.2.1.4.22.1...

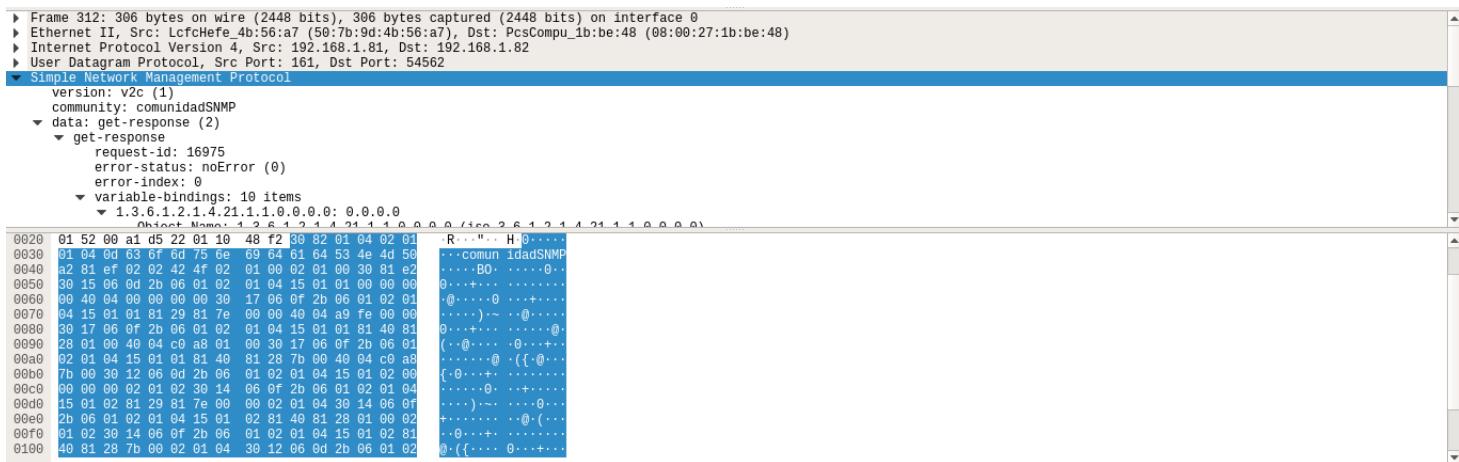


Figura 2.76: Capturas del comando snmptable en Wireshark.

## 2.5.5. Captura de paquetes SNMP formados por el comando snmpwalk

Consulta en consola usando comando **snmpwalk**

```
ESCOM@headhunter:~$ snmpwalk -v 2c -c comunidadSNMP 192.162.1.82 1.3.6.1.2.1.3
```

Figura 2.77: Captura del comando snmpwalk en consola.

Visualización del paquete SNMP en Wireshark

803 140.884152686 192.168.1.82	192.168.1.81	SNMP	91 get-next-request 1.3.6.1.2.1.2.2.1.3
804 140.884617556 192.168.1.81	192.168.1.82	SNMP	93 get-response 1.3.6.1.2.1.2.2.1.3.1
805 140.892029798 192.168.1.82	192.168.1.81	SNMP	92 get-next-request 1.3.6.1.2.1.2.2.2.1.3.1
806 140.892627413 192.168.1.81	192.168.1.82	SNMP	93 get-response 1.3.6.1.2.1.2.2.1.3.2
807 140.900563931 192.168.1.82	192.168.1.81	SNMP	92 get-next-request 1.3.6.1.2.1.2.2.2.1.3.2
808 140.900958892 192.168.1.81	192.168.1.82	SNMP	93 get-response 1.3.6.1.2.1.2.2.1.3.3
809 140.908132161 192.168.1.82	192.168.1.81	SNMP	92 get-next-request 1.3.6.1.2.1.2.2.2.1.3.3
810 140.908562227 192.168.1.81	192.168.1.82	SNMP	93 get-response 1.3.6.1.2.1.2.2.1.3.4
811 140.916154647 192.168.1.82	192.168.1.81	SNMP	92 get-next-request 1.3.6.1.2.1.2.2.2.1.3.4
812 140.916576087 192.168.1.81	192.168.1.82	SNMP	93 get-response 1.3.6.1.2.1.2.2.1.3.5
813 140.923801164 192.168.1.82	192.168.1.81	SNMP	92 get-next-request 1.3.6.1.2.1.2.2.1.3.5
814 140.924225562 192.168.1.81	192.168.1.82	SNMP	95 get-response 1.3.6.1.2.1.2.2.1.4.1

► Frame 805: 92 bytes on wire (736 bits), 92 bytes captured (736 bits) on interface 0
► Ethernet II, Src: PcsCompu_1b:be:48 (08:00:27:1b:be:48), Dst: LfcfHefe_4b:56:a7 (50:7b:9d:4b:56:a7)
► Internet Protocol Version 4, Src: 192.168.1.82, Dst: 192.168.1.81
► User Datagram Protocol, Src Port: 54563, Dst Port: 161
▼ Simple Network Management Protocol
version: v2c (1)
community: comunidadSNMP
▼ data: get-next-request (1)
▼ get-next-request
request-id: 320986
error-status: noError (0)
error-index: 0
variable-bindings: 1 item
▼ 1.3.6.1.2.1.2.2.1.3.1: Value (Null)
Object Name: 1.3.6.1.2.1.2.2.1.3.1 (iso.3.6.1.2.1.2.2.1.3.1)
Value (Null)
0000 50 7b 9d 4b 56 a7 08 00 27 1b be 48 08 00 45 00 P{ KV... ' .H .E .
0010 00 4e 02 97 40 00 80 11 74 14 c8 a8 01 52 c9 a8 ·N .@ ... t ... R .
0020 01 51 d5 23 00 a1 00 3a 54 39 80 30 02 01 01 04 ·0 # ... T9@9 ...
0030 0d 62 6f 6d 75 6e 69 64 61 64 53 4e 4d 50 a1 1c ·comunid adSNMP ..
0040 02 02 7d 56 92 01 00 02 01 00 30 10 38 0e 06 0a ..}V..... ..0.0... .
0050 2b 06 01 02 01 02 02 01 03 01 05 00 +..... ....

Figura 2.78: Capturas del comando snmpwalk en Wireshark.

## 2.6. Conclusiones

### 2.6.1. Conclusión - López Ayala Eric Alejandro

De lo que puedo concluir de esta primera práctica, es que la administración de servicios de red es un tema complejo que abarca muchos tópicos importantes. En este caso lo más relevante fue comprender como nos es útil el protocolo SNMP a la hora de extraer y modificar información de los agentes de una red. La MIB resultó ser una herramienta muy conveniente a la hora de almacenar información de los dispositivos de una manera óptima y efectiva, y mediante la empleación de herramientas como los servicios de snmp tanto en los sistemas operativos de Windows y Linux su acceso fue relativamente sencillo.

Por último esta práctica me ayudó a repasar algunos tópicos importantes de programación web, estructuras de datos y sistemas operativos (multitasking) a la hora de desarrollar nuestro propio sistema de monitoreo de agentes usando django, celery y rrdtool.

### 2.6.2. Conclusión - López Romero Joel

De acuerdo con lo realizado en esta práctica, SNMP fue muy útil, ya que este protocolo nos permite adquirir información de cada dispositivo, mediante una MIB, solamente se debe configurar en este caso para la práctica fue en Windows y Linux. Inicialmente monitoreamos estos agentes en Observium, para entender y analizar cómo funciona SNMP. Toda la teoría vista en clase fue muy precisa y útil para lograr la primera parte, además de todos los ejercicios que realizamos para la adquisición de datos con SNMPGet y SNMPWalk.

Una vez que se logró esta parte de la práctica, realizamos scripts en python con las librerías pySNMP y la herramienta rrdtool, primero realizamos adquisición de datos con un pequeño script en python, después mediante rrdtool, analizamos cómo es que funciona la MIB ya que usa una base de datos round robin y con esto realizamos scripts para graficar ciertos objetos de la MIB en tiempo real.

Todo esto fue útil y necesario para crear nuestra aplicación, la cual administra y monitoriza múltiples agentes, dando gráficas en tiempo real de 5 objetos de la MIB que elegimos. A lo largo de todo este parcial, fue difícil implementar todo lo visto en clase y realizado en laboratorio para crear nuestra aplicación, ya que en un inicio hicimos un multiThreading para manejar cada gráfica de un agente, pero esto no era óptimo, investigando encontramos Celery, una herramienta que nos permite crear procesos asíncronos, así como esto se nos presentaron varios problemas, el más importante fue nuestro entorno de desarrollo, ya que en Windows rrdtool no funciona, en Linux funciona perfectamente pero SNMP fallaba. En general el desarrollo de toda esta práctica fue difícil, pero no imposible además de que aprendí demasiado.

# Bibliografía

- [1] J. M. Huidobro. Snmp: Un protocolo simple de gestión. [Online]. Available: <https://www.coit.es/publicac/publbit/bit102/quees.htm>
- [2] Carlos-vialfa. Protocolo snmp. [Online]. Available: <https://es.ccm.net/contents/280-protocolo-snmp>
- [3] Sosa. Mib - management information base. [Online]. Available: <https://www.tamps.cinvestav.mx/~vjsosa/clases/redes/MIB.pdf>
- [4] M. R. K. McCloghrie. Management information base for network management of tcp/ip-based internets - rfc 1213. [Online]. Available: <https://www.ietf.org/rfc/rfc1213.txt>