

Manual de máquina virtual "A"

Administración de Servicios en Red - ETS 2019

Autor: Roberto Valdez Barba



Contenido:

- Gestor SNMP
- Cliente de correo
- Servidor TFTP
- Servidor FTP

1. Índice

1. Índice	2
2. Descripción General	4
2.1. Preliminares Generales	4
2.1.1. Hardware	4
2.1.2. Software	5
3. SNMP	6
3.1. Introducción	6
3.1.1. Qué es SNMP?	6
3.1.2. Conceptos Básicos	6
3.1.2.1. Imagen: Descripción del funcionamiento de SNMP	7
3.2. Preliminares	8
3.2.1. Hardware	8
3.2.2. Software	8
3.2.3. Permisos	8
3.2.3.1. Configuración del Firewall	8
3.3. Advertencias	9
3.4. Guía de instalación	9
3.4.1. Instalación de Manager SNMP	9
3.4.2. Configuración del SNMP Manager	10
3.4.3. Instalación de agente SNMP	10
3.4.4. Verificación	16
3.4.5. Conclusiones SNMP	17
3.4.6. Referencias	17
4. Cliente de correo	18
5. Servidor FTP	20
5.1. Introducción	20
5.1.1. Qué es FTP?	20
5.1.2. Servidor FTP	22
5.1.3. Cliente FTP	22
5.1.4. Características de FTP	22
5.1.4.1. Acceso anónimo	22
5.1.5. Cliente FTP basado en Web	23
5.1.5.1. Acceso de usuario	23

5.1.6. Modos de conexión del cliente FTP	23
5.1.6.1. Modo activo.	24
5.1.6.2. Modo pasivo.	24
5.2. Preliminares	25
5.3. Advertencias	25
5.4. Guía de instalación	25
5.4.1. Instalación del Servidor FTP	25
5.4.1.1. Comandos del Servidor/Demonio	26
5.4.2. Comandos del servidor/demonio FTP	26
5.4.3. Configuraciones adicionales	26
5.4.3.1. Preparar el directorio	26
5.4.4. Configurando el acceso FTP	28
5.5. Verificación	30
5.7. Referencias	30
6. Servidor TFTP	31
6.1. Introducción	31
6.1.1. Qué es TFTP?	31
6.1.2. Comandos	31
6.2. Preliminares	32
6.3. Advertencias	32
6.4. Guía de instalación	32
6.5. Verificación	32
6.6. Conclusiones TFTP	32
6.7. Referencias	32
6.8. Anexo de configuración	32
6.9. Requerimientos técnicos generales	32
6.10. Conclusiones Generales	34

2. Descripción General

El siguiente manual tiene como finalidad describir las tecnologías detrás de cada uno de servidores implementados en la máquina virtual **"A"**, así como el proceso de instalación, configuración y prueba de los mismos.

Como lo define nuestro documento nuestra máquina virtual la cual denominaremos "Máquina virtual **"A"**" contiene los siguientes elementos:

- Gestor SNMP
- Herramienta de Monitoreo
- Cliente de Correo
- Servidor FTP
- Servidor TFTP

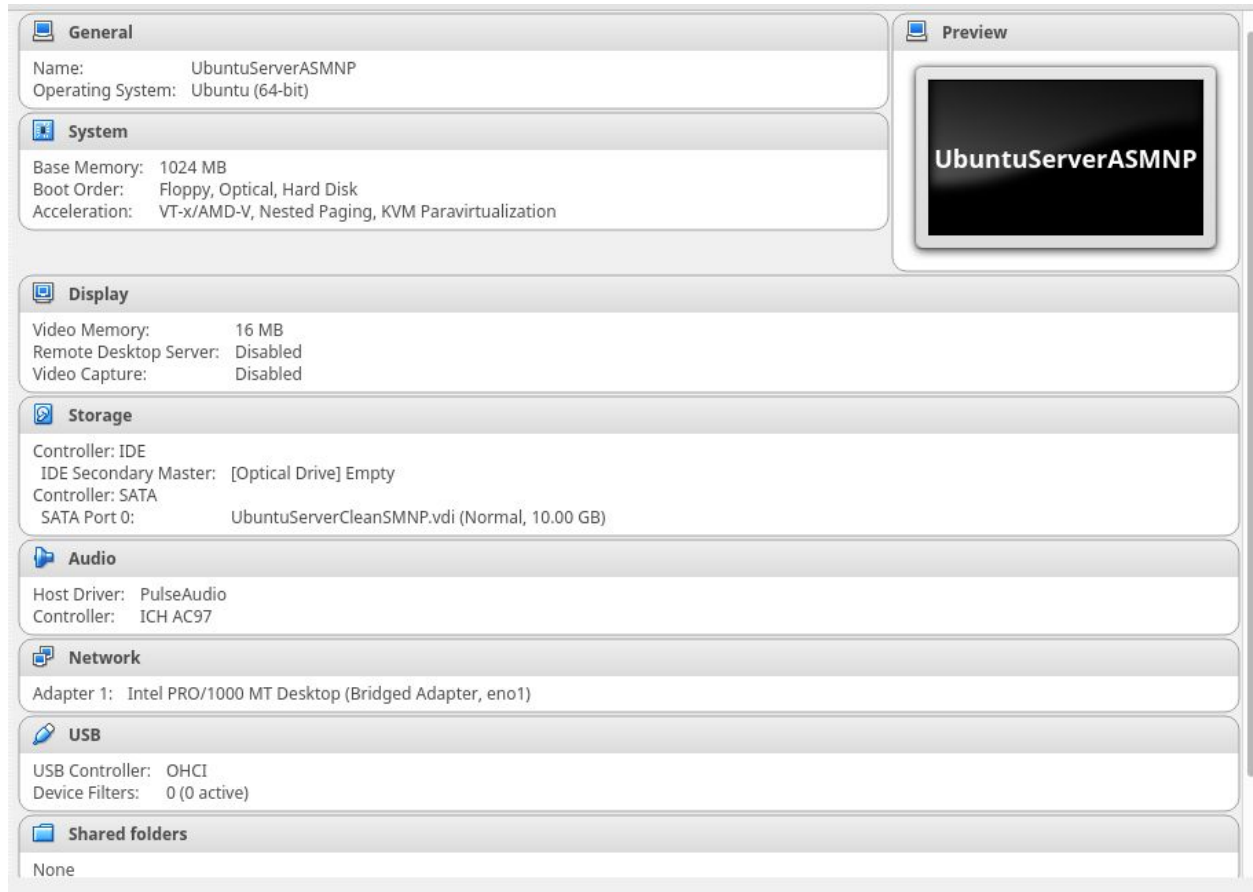
2.1. Preliminares Generales

Para la configuración de los servidores se utilizaron máquinas virtual de Virtualbox, las especificaciones de la máquina virtual del servidor **A** se explicará a continuación.

2.1.1. Hardware

Para poder realizar la instalación del servidor no se necesita de ningún hardware en específico, las características del equipo en donde realizaremos la instalación son las siguientes:

- RAM 512MB
- Disco duro virtual 10 GB
- Tarjeta de red: intel PRO/1000 MT Desktop



Captura de las características de la máquina virtual

2.1.2. Software

El SO Operativo que se eligió para el desarrollo de este examen es **Ubuntu Server 18.04.2 LTS**, se eligió este SO debido a su alta compatibilidad con los paquetes necesarios para los servidores y en su modalidad server para no utilizar muchos recursos, este se puede descargar de la siguiente liga.

<https://ubuntu.com/download/server>

En caso de necesitar ayuda para montar la máquina virtual puede encontrar el manual oficial en el siguiente link: <https://help.ubuntu.com/18.04/installation-guide/>

3. SNMP

NOTA: Debido a la estrecha relación entre el gestor SNMP y el agente SNMP del servidor "B" se explicará la configuración del agente en esta sección.



3.1. Introducción

Parte de la función a desempeñar un administrador de red es reunir información precisa acerca de los diferentes dispositivos que se encuentran distribuidos en la infraestructura de nuestra red, para

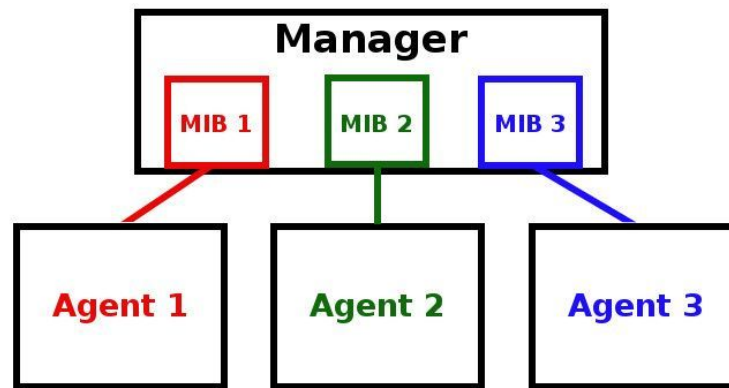
3.1.1. Qué es SNMP?

Las siglas de SNMP significan "*Simple network managment protocol*", este es una forma en que los servidores puede compartir información acerca del estado actual en el que se encuentran, también puede funcionar como medio para modificar valores preconfigurados, El protocolo en si es simple pero la estructura de los programas que implementan SNMP puede resultar muy compleja, a continuación describiremos de manera general cómo se compone el protocolo SNMP.

3.1.2. Conceptos Básicos

SNMP es un protocolo implementado en la capa de aplicación, el protocolo fue creado como una manera de obtener información consisten de de diferentes sistemas, existen diferentes versiones del protocolo SNMP, y una gran cantidad de dispositivos implementan alguna forma de acceso SNMP, la más utilizada es la versión de SNMPv1, aunque esta en diferentes aspectos es insegura.

En general una red de SNMP consiste principalmente de dispositivos que contienen **agentes SNMP**.



3.1.2.1. Imagen: Descripción del funcionamiento de SNMP

Agente SNMP: Se refiere a un programa que pueda recolectar información de una pieza de hardware, organizarla en pequeñas entradas redefinidas y responder a consultas usando el protocolo SNMP.

SNMP Manager: Es el componente que se encarga de realizar las consultas a los agentes, esta máquina normalmente tiene la información sobre todos los dispositivos que cuenta con SNMP, la configuración de este es menor ya que este se encarga solo de solicitar la información a los dispositivos siempre y cuando tengas las credenciales correctas.

Casi todos los comandos que están definidos en el protocolo SNMP están diseñados para enviarse desde el manager entre ellos se encuentran: *GetRequest*, *GetNextRequest*, *GetBulkRequest*, *SetRequest*, *InformRequest*, y *Response*.

Cuando los agentes tienen la información estos la almacenan en un formato que pueda ser consultado por el agente, a esta base de datos se le llama MIB ("Management information base").

3.2. Preliminares

3.2.1. Hardware

- Para poder realizar la instalación del servidor no se necesita de ningún hardware en específico, las características del equipo son las que se definieron en la sección de requerimientos generales.

3.2.2. Software

La instalación de este software puede realizarse en cualquier sistema operativo con base Linux, es nuestro caso este se dará en una **SO Ubuntu Server**. en caso de que desee utilizar algún otra distribución basada en linux puede revisar el siguiente enlace para saber los requerimientos:

<https://docs.oracle.com/cd/E19528-01/819-4740/fwboh/index.html>

3.2.3. Permisos

Para qué la instalación tenga los permisos necesario todos los comandos serán utilizados desde el superusuario o el usuario root, en caso de que se tenga que realizar desde algún otro usuario esto se indicará en la práctica.

3.2.3.1. Configuración del Firewall

Dependiendo la distribución que estemos manejando y la configuración que estemos manejando el firewall de linux puede o no estar habilitado, para saber el estado del mismo podemos utilizar el siguiente comando.

```
sudo ufw status
```

Debemos de tener una respuesta similar a esta

```
Output
Status: active

To Action From
--
OpenSSH ALLOW Anywhere
```


Como podemos ver este se encuentra habilitado y permitiendo conexiones de SSH desde cualquier lugar, para evitar que este pueda causar problemas con los diferentes servidores que utilizaremos a lo largo del manual deshabilitamos el mismo, esto lo podemos hacer a través del siguiente comando:

```
$ sudo ufw disable  
Firewall stopped and disabled on system startup
```

En caso de que necesitemos habilitarlo posteriormente esto lo podemos hacer a través del siguiente comando:

```
$ sudo ufw enable
```

3.3. Advertencias

3.4. Guia de instalacion

3.4.1. Instalación de Manager SNMP

A continuación veremos cómo instalar las utilidades de snmp para la distribución de ubuntu, este nos servirá para poder hacer uso de nuestro script de monitoreo.

Comandos a ejecutar en terminal:

```
sudo apt-get update  
sudo apt-get install snmp snmp-mibs-downloader
```

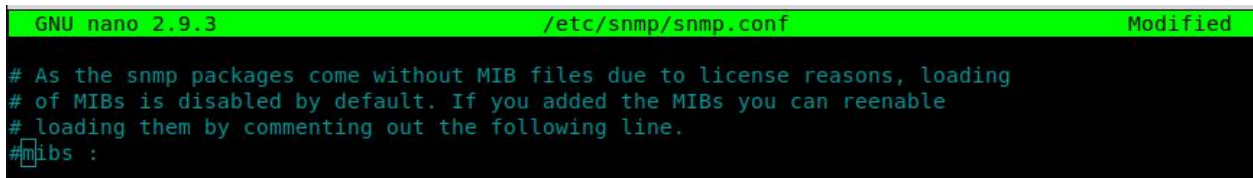
```
robb@tracer:~$ sudo apt-get install snmp snmp-mibs-downloader  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following packages were automatically installed and are no longer required:  
  linux-headers-4.15.0-44 linux-headers-4.15.0-44-generic linux-headers-4.15.0-46  
  linux-headers-4.15.0-46-generic linux-image-4.15.0-44-generic linux-image-4.15.0-46-generic  
  linux-modules-4.15.0-44-generic linux-modules-4.15.0-46-generic  
  linux-modules-extra-4.15.0-44-generic linux-modules-extra-4.15.0-46-generic  
Use 'sudo apt autoremove' to remove them.  
The following additional packages will be installed:  
  smstrip  
The following NEW packages will be installed:  
  smstrip snmp snmp-mibs-downloader  
0 upgraded, 3 newly installed, 0 to remove and 88 not upgraded.  
Need to get 5 330 kB of archives.  
After this operation, 5 914 kB of additional disk space will be used.
```

Captura de la ejecución del comando para la instalación.

3.4.2. Configuración del SNMP Manager

Como se describió en la parte superior la mayoría de la operación del SNMP ocurre en el agente, así que la configuración del manager es muy sencilla. es necesario editar el siguiente archivo para que quede como se muestra en la imagen.

```
sudo nano /etc/snmp/snmpd.conf
```



```
GNU nano 2.9.3 /etc/snmp/snmp.conf Modified
# As the snmp packages come without MIB files due to license reasons, loading
# of MIBs is disabled by default. If you added the MIBs you can reenale
# loading them by commenting out the following line.
#mibs :
```

La razón de esto como lo indica la nota es que hay MIB propietaria, y por defecto esta deshabilitado que se puedan utilizar.

3.4.3. Instalación de agente SNMP

Una vez teniendo el agente funcionando este necesita un equipo para monitorear, como se explica en el documento esta configuración se tiene que realizar en el Server B, decidimos explicarlo aquí para poder entender la relación entre el agente y el gesto.

Comandos a ejecutar en terminal:

```
sudo apt-get update
sudo apt-get install snmpd
```

Una vez descargado el mismo procederemos a editar el archivo en la siguiente ruta:

```
/etc/snmp/snmpd.conf
```

```

GNU nano 2.9.3 /etc/snmp/snmpd.conf
#####
#
# EXAMPLE.conf:
#   An example configuration file for configuring the Net-SNMP agent ('snmpd')
#   See the 'snmpd.conf(5)' man page for details
#
#   Some entries are deliberately commented out, and will need to be explicitly activated
#
#####
#
# AGENT BEHAVIOUR
#
# Listen for connections from the local system only
agentAddress udp:127.0.0.1:161
# Listen for connections on all interfaces (both IPv4 *and* IPv6)
#agentAddress udp:161,udp6:[::1]:161

```

la configuración de este debe de quedar de la siguiente manera:

Las configuraciones que se hacen en la parte de abajo son para poder configurar la red desde la cual se pueden realizar peticiones a la máquina.

```

#####
####
#
# EXAMPLE.conf:
#   An example configuration file for configuring the Net-SNMP agent
#   ('snmpd')
#   See the 'snmpd.conf(5)' man page for details
#
#   Some entries are deliberately commented out, and will need to be
#   explicitly activated
#
#####
####
#
# AGENT BEHAVIOUR
#
# Listen for connections from the local system only
#agentAddress udp:127.0.0.1:161
# Listen for connections on all interfaces (both IPv4 *and* IPv6)
agentAddress udp:161,udp6:[::1]:161

#####
###

```

```
#####
###

#####
####
#
#  SNMPv3 AUTHENTICATION
#
#  Note that these particular settings don't actually belong here.
#  They should be copied to the file /var/lib/snmp/snmpd.conf
#    and the passwords changed, before being uncommented in that file
*only*.
#  Then restart the agent

#  createUser authOnlyUser  MD5 "remember to change this password"
#  createUser authPrivUser  SHA "remember to change this one too"  DES
#  createUser internalUser  MD5 "this is only ever used internally, but
still change the password"

#  If you also change the usernames (which might be sensible),
#  then remember to update the other occurrences in this example config file
to match.

#####
####
#
#  ACCESS CONTROL
#

#  system + hrSystem

groups only
#view  systemonly  included  .1.3.6.1.2.1.1
#view  systemonly  included  .1.3.6.1.2.1.25.1
view systemview  included .1.3.6.1.2.1.25

#  Full access from the
```

```

local host
rocommunity public localhost
# Default access to basic
system info
rocommunity public default
# rocommunity6 is for
IPv6
rocommunity6 public default -V systemonly
# Full access from an
example network
# Adjust this network
address to match your local
# settings, change the
community string,
# and check the
'agentAddress' setting above
rocommunity secret 192.168.1.0/24
# Full read-only access
for SNMPv3
rouser authOnlyUser
# Full write access for
encrypted requests
# Remember to activate
the 'createUser' lines above
#rwuser authPrivUser priv
# It's no longer typically necessary to use the full
'com2sec/group/access' configuration
# r[ow]user and r[ow]community, together with suitable views, should cover
most requirements

#####
####
#
# SYSTEM INFORMATION
#
# Note that setting these values here, results in the corresponding MIB

```

```

objects being 'read-only'
# See snmpd.conf(5) for more details
sysLocation      Sitting on the Dock of the Bay
sysContact       Me <me@example.org>

# Application + End-to-End
layers
sysServices      72

#
# Process Monitoring
#
# At least one 'mountd' process
proc mountd
# No more than 4 'ntalkd' processes - 0 is
OK
proc ntalkd      4
# At least one 'sendmail' process, but no
more than 10
proc sendmail 10 1

# Walk the UCD-SNMP-MIB::prTable to see the resulting output
# Note that this table will be empty if there are no "proc" entries in the
snmpd.conf file

#
# Disk Monitoring
#
# 10MBs required on root disk, 5% free on
/var, 10% free on all other disks
disk      /      10000
disk      /var   5%
includeAllDisks 10%

# Walk the UCD-SNMP-MIB::dskTable to see the resulting output
# Note that this table will be empty if there are no "disk" entries in the
snmpd.conf file

#
# System Load
#
# Unacceptable 1-, 5-, and 15-minute load

```

```

averages
load    12 10 5

# Walk the UCD-SNMP-MIB::laTable to see the resulting output
# Note that this table *will* be populated, even without a "load" entry in
the snmpd.conf file

#####
####
#
# ACTIVE MONITORING
#

trapsink    localhost public    # send SNMPv1 traps
#trap2sink  localhost public    # send SNMPv2c traps
#informsink localhost public    # send SNMPv2c INFORMs

# Note that you typically only want *one* of these three lines
# Uncommenting two (or all three) will result in multiple copies of each
notification.

#
# Event MIB - automatically generate alerts
#
# Remember to activate the 'createUser'
lines above
iquerySecName internalUser
rouser        internalUser
# generate traps on UCD error conditions
defaultMonitors yes
# generate traps on linkUp/Down
linkUpDownNotifications yes

#####

```

```
####
#  EXTENDING THE AGENT
#  extend      test1    /bin/echo Hello, world!
#  extend-sh   test2    echo Hello, world! ; echo Hi there ; exit 35
#extend-sh    test3    /bin/sh /tmp/shtest
#
#  AgentX Sub-agents
#                                     #  Run as an AgentX master agent
master        agentx
#                                     #  Listen for network
```

Una vez hecha la configuración de nuestro SNMPD tenemos que reiniciar el agente para que entre en vigor la configuración,

```
sudo service snmpd restart
```

3.4.4. Verificación

Para saber qué nuestro agente se encuentra correctamente en funcionamiento ejecutaremos un query, los queries se componen de la siguiente forma:

```
snmpwalk -v2c -c public 192.168.1.67 .1.3.6.1.2.1.2.2.1.16
```

snmpwalk -v2c : Indica la versión de SNMP que se está utilizando

public: indica la comunidad que se utiliza

IP: Dirección a la que se le quiere hacer la petición

OID: Información que se requiere del mismo

Si todo está funcionando correctamente obtendremos en pantalla una respuesta similar a la siguiente:

```
robb@userver:~/monitoreo$ snmpwalk -v2c -c public 192.168.1.67 .1.3.6.1.2.1.2.2.1.16
IF-MIB::ifOutOctets.1 = Counter32: 11901018
IF-MIB::ifOutOctets.2 = Counter32: 301692531
IF-MIB::ifOutOctets.3 = Counter32: 0
robb@userver:~/monitoreo$
```

Captura de resultado de query al agente

3.4.5. Conclusiones SNMP

Los gestores SNMP son la mano derecha de la gestión de los servicios en red, gracias a ellos podemos conocer el estado de cada uno de nuestros dispositivos en la red, saber qué es lo que necesitan y automatizar todo el proceso, además de qué gracias a su amplio uso hoy en día no solo aplica a computadoras y servidor, puede ser implementado para todo en el “ Internet de las cosas”

3.4.6. Referencias

DigitalOcean. “An Introduction to SNMP (Simple Network Management Protocol).” *DigitalOcean*,

DigitalOcean, 18 July 2017,

www.digitalocean.com/community/tutorials/an-introduction-to-snmp-simple-network-management-protocol.

Weidenbacher, Nik, and Nik Weidenbacher. “Bash and Net-SNMP: a Low Budget, High

Frequency SNMP Poller.” *Packet Pushers*, 16 Oct. 2013,

packetpushers.net/bash-and-net-snmp-a-low-budget-high-frequency-snmp-poller/.

4. Cliente de correo

Para la configuración del cliente de correo se nos solicitó que funcionara con **SNMPD_CUSTOM**

```
#!/usr/bin/env python
# encoding: utf-8

import smtplib
import asyncore

class CustomSMTPServer(smtplib.SMTPServer):

    def process_message(self, peer, mailfrom, rcpttos, data):
        print ('Receiving message from:', peer)
        print ('Message addressed from:', mailfrom)
        print ('Message addressed to  :', rcpttos)
        print ('Message length        :', len(data))
        print ('Message data             :', data)
        return

server = CustomSMTPServer(('192.168.1.67', 1025), None)

asyncore.loop()
```

Para poder enviar correos al SNMPD_construimos un SCRIPT que se encargará de realizar una conexión

```
#!/usr/bin/env python
# encoding: utf-8
#smtpd_senddata.py

import smtplib
import email.utils
from email.mime.text import MIMEText
```

```
# Create the message
msg = MIMEText('This is the body of the message.')
msg['To'] = email.utils.formataddr(('Recipient','recipient@example.com'))
msg['From'] = email.utils.formataddr(('Author','author@example.com'))
msg['Subject'] = 'Simple test message'

server = smtplib.SMTP('127.0.0.1', 1025)
server.set_debuglevel(True) # show communication with the server
try:
    server.sendmail('author@example.com',['recipient@example.com'],
msg.as_string())
finally:
    server.quit()
```

5. Servidor FTP



5.1. Introducción

5.1.1. Qué es FTP?

Las siglas de FTP vienen de “File Transfer Protocol” , Protocolo para la transferencia de archivos, como su nombre lo indica es un protocolo para la transferencia de archivos en una red TCP, basado en la arquitectura cliente servidor, desde un equipo cliente se puede conectar a un servidor para descargar archivos desde él o para enviarle archivos, independientemente del sistema operativo utilizado en cada equipo.

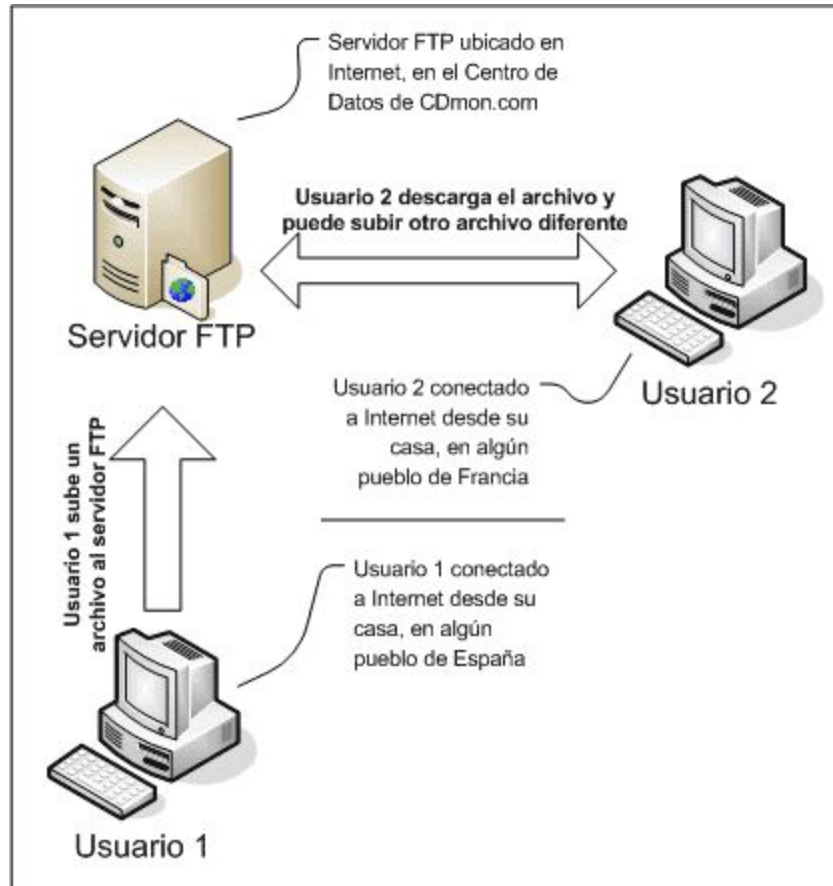


Diagrama del proceso de funcionamiento de un FTP
Observamos que intervienen tres elementos:

Descripción de los elementos:

El servidor FTP: Donde subiremos / descargamos los archivos.

Usuario 1: Es el usuario que en este ejemplo, sube un archivo al servidor FTP.

Usuario 2: Es el usuario que en este ejemplo, se descarga el archivo subido por el usuario 1 y a continuación sube otro archivo.

En el modelo, el intérprete de protocolo (PI) de usuario inicia la conexión de control en el puerto 21. Las órdenes FTP estándar las genera el PI de usuario y se transmiten al proceso servidor a través de la conexión de control. Las respuestas estándar se envían desde la PI del servidor hasta la PI de usuario por la conexión de control como respuesta a las órdenes.

Estas órdenes FTP especifican parámetros para la conexión de datos (puerto de datos, modo de transferencia, tipo de representación y estructura) y la naturaleza de la operación sobre el sistema de archivos (almacenar, recuperar, añadir, borrar, etc.). El proceso de transferencia de datos (DTP) de usuario u otro proceso en su lugar, debe esperar a que el servidor inicie la conexión al puerto de datos especificado (puerto 20 en modo activo o estándar) y transferir los datos en función de los parámetros que se hayan especificado.

También hay que destacar que **la conexión de datos es bidireccional**, es decir, se puede usar simultáneamente para enviar y para recibir, y no tiene por qué existir todo el tiempo que dura la conexión FTP. Pero tenía en sus comienzos un problema, y era la localización de los servidores en la red. Es decir, el usuario que quería descargar algún archivo mediante trump debía conocer en qué máquina estaba ubicado. La única herramienta de búsqueda de información que existía era Gopher, con todas sus limitaciones.

5.1.2. Servidor FTP

Un servidor FTP es un programa especial que se ejecuta en un equipo servidor normalmente conectado a Internet (aunque puede estar conectado a otros tipos de redes, LAN, MAN, etc.). Su función es permitir el intercambio de datos entre diferentes servidores/ordenadores.

5.1.3. Cliente FTP

Un cliente FTP emplea el FTP para conectarse a un servidor FTP para transferir archivos a un alojamiento.

Algunos clientes de FTP básicos vienen integrados en los sistemas operativos, incluyendo Windows, DOS, Linux y Unix. Sin embargo, hay disponibles clientes con más funcionalidades, habitualmente en forma de shareware/freeware para Windows y como software libre para sistemas de tipo Unix. Muchos navegadores recientes también llevan integrados clientes FTP (aunque un cliente FTP trabajará mejor para FTP privadas que un navegador).

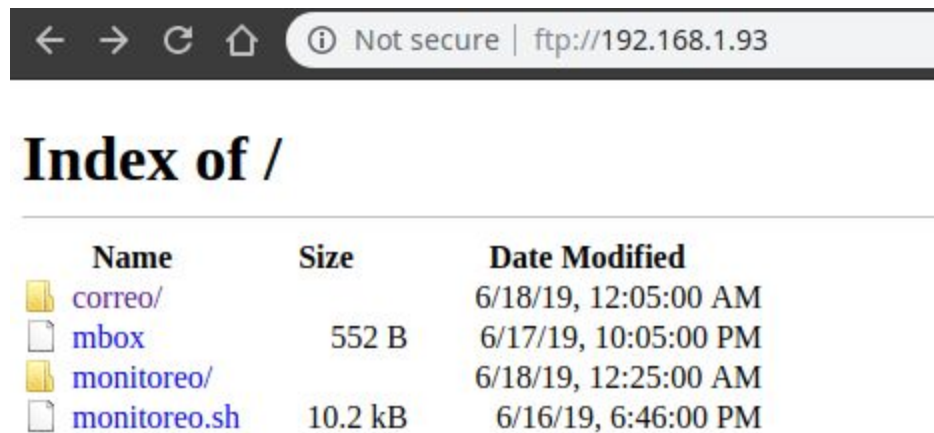
5.1.4. Características de FTP

5.1.4.1. Acceso anónimo

Los servidores FTP anónimos ofrecen sus servicios libremente a todos los usuarios, permiten acceder a sus archivos sin necesidad de tener un 'USER ID' o una cuenta de usuario. Es la manera más cómoda fuera del servicio web de permitir que todo el mundo tenga acceso a cierta información sin que para ello el administrador de un sistema tenga que crear una cuenta para cada usuario.

5.1.5. Cliente FTP basado en Web

Un «cliente FTP basado en Web» no es más que un cliente FTP al cual podemos acceder a través de nuestro navegador web sin necesidad de tener otra aplicación para ello. El usuario se conecta mediante HTTP a un servidor web, y el servidor web se conecta mediante FTP al servidor de archivos. El servidor web actúa de intermediario haciendo pasar la información desde el servidor FTP en los puertos 20 y 21 hacia el puerto 80 HTTP que ve el usuario.



Representación de como acceder al servidor mediante navegador

5.1.5.1. Acceso de usuario

Si se desea tener privilegios de acceso a cualquier parte del sistema de archivos del servidor FTP, de modificación de archivos existentes, y de posibilidad de subir nuestros propios archivos, generalmente se suele realizar mediante una cuenta de usuario. En el servidor se guarda la información de las distintas cuentas de usuario que pueden acceder a él, de manera que para iniciar una sesión FTP debemos introducir una autenticación (en inglés: login) y una contraseña (en inglés: password) que nos identifica unívocamente.

5.1.6. Modos de conexión del cliente FTP

FTP admite dos modos de conexión del cliente. Estos modos se denominan activo (o Estándar, o PORT, debido a que el cliente envía comandos tipo PORT al servidor por el canal de control al establecer la conexión) y pasivo (o PASV, porque en este caso envía comandos tipo PASV). Tanto en el modo Activo como en el modo Pasivo, el cliente establece una conexión con el servidor mediante el puerto 21, que establece el canal de control.

5.1.6.1. Modo activo.

En modo Activo, el servidor siempre crea el canal de datos en su puerto 20, mientras que en el lado del cliente el canal de datos se asocia a un puerto aleatorio mayor que el 1024. Para ello, el cliente manda un comando PORT al servidor por el canal de control indicándole ese número de puerto, de manera que el servidor pueda abrirle una conexión de datos por donde se transferirán los archivos y los listados, en el puerto especificado.

Lo anterior tiene un grave problema de seguridad, y es que la máquina cliente debe estar dispuesta a aceptar cualquier conexión de entrada en un puerto superior al 1024, con los problemas que ello implica si tenemos el equipo conectado a una red insegura como Internet. De hecho, los cortafuegos que se instalen en el equipo para evitar ataques seguramente rechazarán esas conexiones aleatorias. Para solucionar esto se desarrolló el modo pasivo.

5.1.6.2. Modo pasivo.

Cuando el cliente envía un comando PASV sobre el canal de control, el servidor FTP le indica por el canal de control, el puerto (mayor a 1024 del servidor. Ejemplo:2040) al que debe conectarse el cliente. El cliente inicia una conexión desde el puerto siguiente al puerto de control (Ejemplo: 1036) hacia el puerto del servidor especificado anteriormente (Ejemplo: 2040).[5]

Antes de cada nueva transferencia tanto en el modo Activo como en el Pasivo, el cliente debe enviar otra vez un comando de control (PORT o PASV, según el modo en el que haya conectado), y el servidor recibirá esa conexión de datos en un nuevo puerto (aleatorio si es en modo pasivo o por el puerto 20 si es en modo activo).

5.2. Preliminares

5.3. Advertencias

5.4. Guia de instalacion

5.4.1. Instalación del Servidor FTP

Para nuestro servidor utilizaremos el servidor VSFTPD, este está diseñado para sistemas basados en UNIX. En el SO operativo donde realizaremos la instalación se encuentra entre los repositorios oficiales, comenzaremos con la actualización y posteriormente instalar el demonio.

```
#Este comando instalará el ftp
sudo apt-get update
sudo apt-get install vsftpd
```

```
robb@userver:~$ sudo apt-get install vsftpd
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  vsftpd
0 upgraded, 1 newly installed, 0 to remove and 85 not upgraded.
Need to get 115 kB of archives.
After this operation, 334 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu bionic/main amd64 vsftpd amd64 3.0.3-9build1 [115 kB]
Fetched 115 kB in 1s (223 kB/s)
Preconfiguring packages ...
Selecting previously unselected package vsftpd.
(Reading database ... 69302 files and directories currently installed.)
Preparing to unpack .../vsftpd_3.0.3-9build1_amd64.deb ...
Unpacking vsftpd (3.0.3-9build1) ...
Processing triggers for ureadahead (0.100.0-20) ...
Setting up vsftpd (3.0.3-9build1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/vsftpd.service → /lib/systemd/system/vsftpd.service.
Processing triggers for systemd (237-3ubuntu10.19) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for ureadahead (0.100.0-20) ...
robb@userver:~$
```

A continuación procederemos a realizar una copia de la configuración original del mismo para poder realizar nuestra configuración en blanco:

```
sudo cp /etc/vsftpd.conf /etc/vsftpd.conf.orig
```

5.4.1.1. Comandos del Servidor/Demonio

Por defecto vsftpd inicia en modo stand-alone, este modo cuenta con su propio demonio en **/etc/init.d/vsftpd/**, este cuenta con las siguientes operaciones para controlarlo:

5.4.2. Comandos del servidor/demonio FTP

Comando	Descripcion
start / stop	Usado para iniciar o detener el servidor
status	provee información del estado actual del servidor
restart	Alternativa a usar start o stop, si esta detenido lo iniciara
reload	La diferencia entre reload y restart es que con reload no se detiene el servidor, solo se recarga la configuración.

5.4.3. Configuraciones adicionales

5.4.3.1. Preparar el directorio

Para este tutorial, vamos a crear un usuario, pero es posible que ya tenga un usuario que necesite acceso a FTP. Nos encargaremos de preservar el acceso de un usuario existente a sus datos en las instrucciones que siguen. Aun así, le recomendamos que comience con un nuevo usuario hasta que haya configurado y probado su configuración.

```
sudo adduser sammy
```

Por lo general, FTP es más seguro cuando los usuarios están restringidos a un directorio específico. vsftpd logra esto con las jaulas **chroot**. Cuando chroot está habilitado para usuarios locales, están restringidos a su directorio de inicio por defecto. Sin embargo, debido a la forma en que vsftpd asegura el directorio, el usuario no debe poder escribirlo. Esto está bien para un nuevo usuario que solo debe conectarse a través de FTP, pero un usuario existente puede necesitar escribir en su carpeta de inicio si también tiene acceso de shell.

En este ejemplo, en lugar de eliminar los privilegios de escritura del directorio de inicio, crearemos un directorio ftp para que sirva como chroot y un directorio de files escritura para contener los archivos reales.

Cree la carpeta ftp , establezca su propiedad y asegúrese de eliminar los permisos de escritura con los siguientes comandos:

```
sudo mkdir / home / sammy / ftp
sudo chown nobody: nogroup / home / sammy / ftp
sudo chmod aw / home / sammy / ftp
```

Vamos a verificar los permisos:

```
sudo ls -la / home / sammy / ftp
```

```
Salida
total 8
4 dr-xr-xr-x 2 nadie nogrupa 4096 Ago 24 21:29.
4 drwxr-xr-x 3 sammy sammy 4096 24 de agosto 21:29 ..
```

A continuación, crearemos el directorio donde se pueden cargar los archivos y asignaremos la propiedad al usuario:

```
sudo mkdir / home / sammy / ftp / files
sudo chown sammy : sammy / home / sammy / ftp / files
```

Una verificación de permisos en el directorio de files debe devolver lo siguiente:

```
sudo ls -la / home / sammy / ftp
Salida
total 12
dr-xr-xr-x 3 nadie nogrupa 4096 Ago 26 14:01.
drwxr-xr-x 3 sammy sammy 4096 26 de agosto 13:59 ..
drwxr-xr-x 2 sammy sammy 4096 26 de agosto 14:01 archivos
```

Finalmente, agregaremos un archivo test.txt para usar cuando test.txt más adelante:

```
echo "archivo de prueba vsftpd" | sudo tee / home / sammy
/ftp/files/test.txt
```

Ahora que hemos asegurado el directorio ftp y hemos permitido al usuario acceder al directorio de files , dirigiremos nuestra atención a la configuración.

5.4.4. Configurando el acceso FTP

Estamos planeando permitir que un solo usuario con una cuenta de shell local se conecte con FTP. Las dos configuraciones clave para esto ya están establecidas en vsftpd.conf . Comience abriendo el archivo de configuración para verificar que la configuración de su configuración coincida con la siguiente:

```
sudo nano /etc/vsftpd.conf
```

```
/etc/vsftpd.conf
. . . # Allow anonymous FTP? (Disabled by default). anonymous_enable=NO #
# Uncomment this to allow local users to log in. local_enable=YES . . .
```

A continuación tendremos que cambiar algunos valores en el archivo. Para permitir que el usuario cargue archivos, descomentamos la configuración write_enable para que tengamos:

```
/etc/vsftpd.conf
. . . write_enable= YES . . .
```

También descomentaremos el chroot para evitar que el usuario conectado a FTP acceda a archivos o comandos fuera del árbol de directorios.

```
/etc/vsftpd.conf
. . . chroot_local_user= YES . . .
```

user_sub_token un user_sub_token para insertar el nombre de usuario en nuestra local_root directory para que nuestra configuración funcione para este usuario y para cualquier usuario futuro que pueda agregarse.

```
/etc/vsftpd.conf
user_sub_token=$USER local_root=/home/$USER/ftp
```

Limitaremos el rango de puertos que se pueden usar para FTP pasivo para asegurarnos de que haya suficientes conexiones disponibles:

```
/etc/vsftpd.conf
pasv_min_port=40000 pasv_max_port=50000
```

Nota: abrimos previamente los puertos que configuramos aquí para el rango de puertos pasivos.

Si cambia los valores, asegúrese de actualizar la configuración de su firewall.

Como solo planeamos permitir el acceso a FTP caso por caso, configuraremos la configuración para que el usuario solo tenga acceso cuando se agrega explícitamente a una lista en lugar de hacerlo de manera predeterminada:

```
/etc/vsftpd.conf
userlist_enable=YES userlist_file=/etc/vsftpd.userlist userlist_deny=NO
```

userlist_deny alterna la lógica. Cuando se establece en "SÍ", a los usuarios de la lista se les niega el acceso a FTP. Cuando se establece en "NO", solo se permite el acceso a los usuarios de la lista. Cuando haya terminado de realizar el cambio, guarde y salga del archivo.

Finalmente, crearemos y agregaremos nuestro usuario al archivo. Usaremos la bandera -a para añadir al archivo:

```
echo " sammy " | sudo tee -a /etc/vsftpd.userlist
```

Comprueba que se haya añadido como esperabas:

```
cat /etc/vsftpd.userlist
Salida
sammy
```

Reinicie el demonio para cargar los cambios de configuración:

```
sudo systemctl restart vsftpd
```

5.5. Verificación

```
robb@tracer:/etc/snmp$ ftp ftp.ets.net
Connected to ftp.ets.net.
220 (vsFTPD 3.0.3)
Name (ftp.ets.net:robb): robb
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
drwxrwxr-x   2 1000    1000          4096 Jun 18 05:05 correo
drwxrwxr-x   2 1000    1000          4096 Jun 20 15:15 monitoreo
-rwxrwxr-x   1 1000    1000        10442 Jun 16 23:46 monitoreo.sh
226 Directory send OK.
ftp> exit
221 Goodbye.
```

5.7. Referencias

DigitalOcean. "How To Set Up Vsftpd for a User's Directory on Ubuntu 16.04." *DigitalOcean*,

DigitalOcean, 6 July 2018,

www.digitalocean.com/community/tutorials/how-to-set-up-vsftpd-for-a-user-s-directory-on-ubuntu-16-04.

6. Servidor TFTP

6.1. Introducción

6.1.1. Qué es TFTP?

TFTP significa Trivial File Transfer Protocol. Se utiliza principalmente para la transferencia de archivos automatizados entre máquinas sobre el puerto UDP 69. En un entorno de VoIP el TFTP se utiliza para cargar los archivos de firmware en gateways, teléfonos y otros equipos.

6.1.2. Comandos

```
TFTP <nombre_host>  
lleva a la línea de órdenes donde se pueden introducir las siguientes  
subórdenes:  
  
Connect <host>  
especifica el ID del host de destino  
Mode <ascii/binary>  
especifica el tipo del modo de transferencia  
Get <nombre_fichero remoto> [<nombre_fichero local>]  
recupera un fichero  
Put <nombre_fichero remoto> [<nombre_fichero local>]  
almacena un fichero  
Verbose  
cambia a modo verboso, que muestra información adicional durante la  
transferencia de ficheros, on o off  
Quit  
sale de TFTP  
Para una lista completa de estas órdenes, ver la guía de usuario de la  
implementación particular de TFTP.
```

6.2. Preliminares

6.3. Advertencias

6.4. Guia de instalacion

```
sudo apt-get install xinetd tftpd tftp
```

6.5. Verificación

6.6. Conclusiones TFTP

6.7. Referencias

6.8. Anexo de configuración

En esta sección manejaremos configuraciones adicionales que si bien no son necesaria para el funcionamiento de los programas/servidores anteriormente mencionados fueron en las pruebas de funcionamiento de los mismos.

6.9. Requerimientos técnicos generales

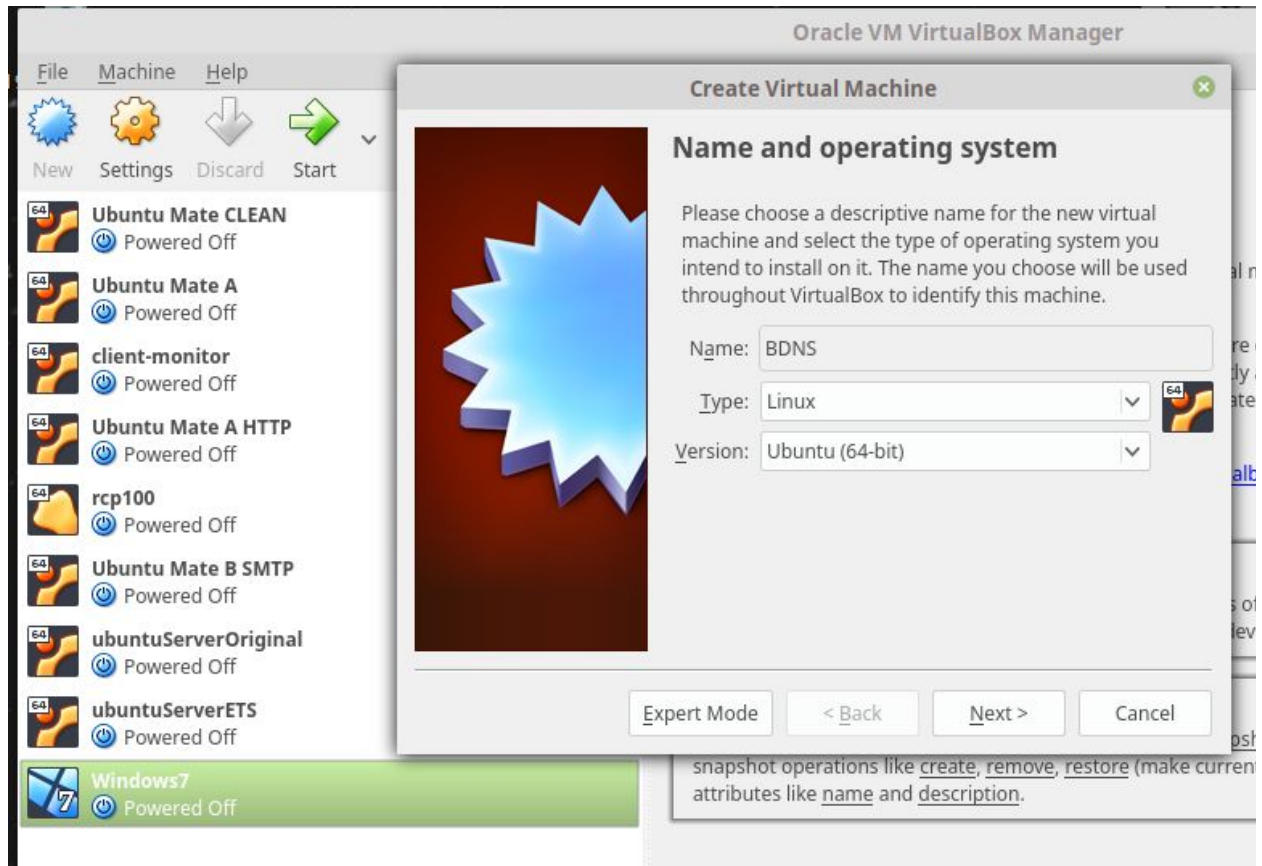
Como se menciona en el texto de la prueba en esta ocasión tendremos que importar los .VDI de la máquina virtual que ya contiene los servidores pre instalados.

Para poder incorporar los discos abriremos el programa virtualbox (Suponemos ya se encuentra instalado), en caso contrario los podemos encontrar en el siguiente enlace

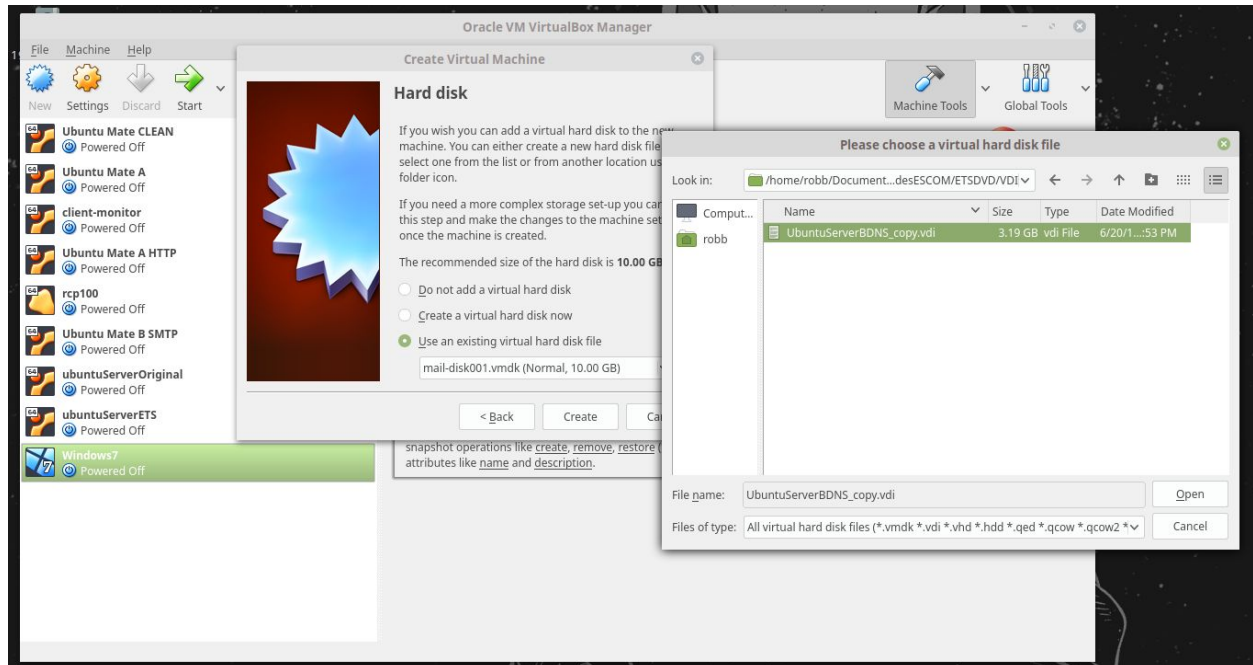
https://www.virtualbox.org/wiki/Linux_Downloads

(Dentro de la misma pagina se encuentra el manual de instalación dependiendo el tipo de SO)

Seleccionaremos que queremos hacer una nueva máquina virtual, de tipo Linux



Cuando nos pregunte si deseamos crear un nuevo disco duro, seleccionaremos qué no y abriremos la ubicación de nuestro VDI.



Continuaremos dando click a finalizar, y e iniciando nuestra máquina virtual de manera normal, si la configuración de todo lo demás se hizo normal

6.10. Conclusiones Generales

Las servidores que se instalaron juegan un papel vital en las comunicaciones del día a día, debo destacar que la parte que causa más valor a mi día a día es la de configurar SNMP, dado que esta implementado en gran cantidad de dispositivos este me permite tener un control sobre la red y los requerimientos, ya sea de los equipos como de los usuarios.