

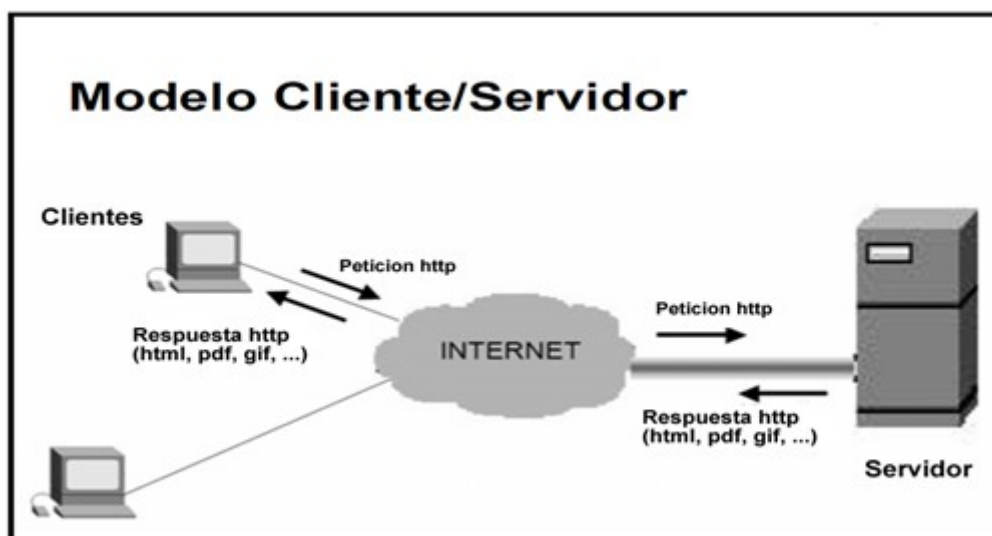
# Programación de Servidores

## Unidad I

### Servidor

Es un dispositivo o infraestructura informática que es parte de una red y su tarea es almacenar, distribuir y suministrar información. Los servidores funcionan basándose en el modelo **cliente-servidor**: El cliente puede ser tanto una computadora como una aplicación que requiere información del servidor para funcionar. El servidor ofrecerá la información demandada por el cliente siempre y cuando el cliente esté autorizado.

La arquitectura cliente-servidor es un modelo de aplicación distribuida en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Un cliente realiza peticiones a otro programa, el servidor, quien le da respuesta. Esta idea también se puede aplicar a programas que se ejecutan sobre una sola computadora, aunque es más ventajosa en un sistema operativo multiusuario distribuido a través de una red de computadoras.



Los servidores pueden ser físicos o virtuales.

En el caso del físico, se trata de un dispositivo de hardware, también conocido como host (anfitrión) o servidor dedicado, normalmente es una computadora en forma de torre o rack integrada a una red de nodos de comunicación que puede mantener diferentes recursos compartidos en el mismo sistema operativo.

En cambio, un servidor virtual (VPS, Virtual Private Server) es un servicio de alojamiento web que se obtiene dividiendo un servidor físico en varios servidores virtuales a través de software de virtualización, haciendo que cada uno de ellos cuente con recursos dedicados y esté aislado de los demás.

Los servicios que prestan los servidores son requeridos continuamente y, por tanto, la mayoría de los servidores nunca se apagan. Si un servidor dejara de funcionar, eso puede causar muchos problemas a los usuarios, por eso los servidores suelen estar configurados con redundancia y alta tolerancia a fallos.

## Tipos de servidores

Dentro del mundo de los servidores hay una gran variedad. Los más usados y/o conocidos son los siguientes:

**Servidor Web:** almacena y organiza el contenido de las páginas web y se lo proporciona al usuario a través del navegador web del usuario. La transmisión de los datos se suele realizar con http (HyperText Transfer Protocol). El http se encarga de transmitir la información de la WWW (World Wide Web), estableciendo unos criterios semánticos y sintácticos para que el ordenador solicitante y el que contiene la información tengan un idioma común en el que comunicarse. Los servidores web más conocidos son: Apache web server; Nginx; y Microsoft.

**DNS:** son las siglas anglosajonas de Domain Name Server, aunque la “S” también puede referirse a service o space sin que cambie el significado ni la función del servidor. El dominio es el nombre que recibe una página web. El servidor DNS se encarga de relacionar una dirección de dominio (www.ejemplodominio.es) con su dirección IP correspondiente. El proceso de enlace con la página web funciona de la siguiente forma. Primero un usuario inserta la dirección de dominio en su motor de búsqueda, esta petición llega a un servidor DNS el cual transforma dicho dominio en su dirección IP, esta dirección IP es reconocida por el servidor web y este último proporciona el contenido de la web al usuario. Además, el dominio web es importante para la imagen de la empresa. El dominio se marca con el final de la dirección web (.com, .es, nl, .net, .org, ...). Si una empresa quiere evitar que en otra parte del mundo haya otra empresa con el mismo nombre y tenga la misma dirección web pero con otro final. Por ejemplo, www.empresa.com y www.empresa.es. Entonces, para que todo dominio con ese nombre redirija a una empresa, dicha empresa debe adquirir y conservar todos los dominios.

**Proxy Server:** es un servidor de puerta de entrada. Se encarga de conectar una red cliente (navegador web o aplicación) con un sistema externo para que se puede llevar a cabo la solicitud de conexión, mejor rendimiento y accesibilidad. Es decir, la conexión se beneficia del caché que almacena, haciendo que una página ya visitada por el usuario se cargue más rápido y reduciendo el ancho de banda de la red. Por ejemplo, cuando se suele mandar un tipo de factura a ciertos destinatarios, el sistema propone autocompletar uno o varios campos con los datos de aquellos destinatarios recurrentes.

**Servidor de correo electrónico:** se encarga del flujo de correo electrónico de los usuarios, permitiendo que se almacene, envíe, reciba y reenvíe los e-mails.

**Servidor FTP:** el nombre procede de File Transfer Protocol, también conocido como protocolo de transferencia de archivos en español. Sirve para transferir archivos entre un cliente y un servidor. De esta forma, el servidor puede recibir archivos del cliente y le posibilita la descarga de los archivos a los clientes. Para mejorar la velocidad, se usa SFTP el cual también ofrece un servicio de seguridad encriptada.

Estos son los tipos de servidores más conocidos, pero no por ello los únicos. Dentro del mercado existen servidores que ofrecen una amplia variedad de servicios, desde operatividad en el sector de los videojuegos a chat empresarial.

Por otra parte, también están lo que comúnmente se conoce como **servidor de bases de datos**. Aunque se le conozca como “servidor”, no es un servidor, sino un tipo de software que está instalado en un servidor. Una base de datos sirve para almacenar grandes cantidades de información. De esta forma, permite que otros programas o clientes accedan a esa información siempre que estén autorizados.

## Modelo Cliente-Servidor

La red cliente-servidor es una red de comunicaciones en la cual los clientes están conectados a un servidor, en el que se centralizan los diversos recursos y aplicaciones con que se cuenta; y que los pone a disposición de los clientes cada vez que estos son solicitados. Esto significa que todas las gestiones que se realizan se concentran en el servidor, de manera que en él se disponen los requerimientos provenientes de los clientes que tienen prioridad, los archivos que son de uso público y los que son de uso restringido, los archivos que son de sólo lectura y los que, por el contrario, pueden ser modificados, etc. Este tipo de red puede utilizarse conjuntamente en caso de que se este utilizando en una red mixta.

## Características

En la arquitectura C/S el remitente de una solicitud es conocido como cliente.

- Es quien inicia solicitudes o peticiones, tienen por tanto un papel activo en la comunicación (dispositivo maestro o amo).
- Espera y recibe las respuestas del servidor.
- Por lo general, puede conectarse a varios servidores a la vez.
- Normalmente interactúa directamente con los usuarios finales mediante una interfaz gráfica de usuario.

Al receptor de la solicitud enviada por el cliente se conoce como servidor.

- Al iniciarse esperan a que lleguen las solicitudes de los clientes, desempeñan entonces un papel pasivo en la comunicación (dispositivo esclavo).
- Tras la recepción de una solicitud, la procesan y luego envían la respuesta al cliente.
- Por lo general, acepta las conexiones de un gran número de clientes (en ciertos casos el número máximo de peticiones puede estar limitado).

Características generales de la arquitectura:

- El Cliente y el Servidor pueden actuar como una sola entidad y también pueden actuar como entidades separadas, realizando actividades o tareas independientes.

- Las funciones de Cliente y Servidor pueden estar en plataformas separadas, o en la misma plataforma.
- Cada plataforma puede ser escalable independientemente. Los cambios realizados en las plataformas de los Clientes o de los Servidores, ya sean por actualización o por reemplazo tecnológico, se realizan de una manera transparente para el usuario final.
- La interrelación entre el hardware y el software están basados en una infraestructura poderosa, de tal forma que el acceso a los recursos de la red no muestra la complejidad de los diferentes tipos de formatos de datos y de los protocolos.
- Su representación típica es un centro de trabajo (PC), en donde el usuario dispone de sus propias aplicaciones de oficina y sus propias bases de datos, sin dependencia directa del sistema central de información de la organización.

## Componentes básicos del modelo Cliente-Servidor

En esta aproximación, y con el objetivo de definir y delimitar el modelo de referencia de una arquitectura Cliente/Servidor, se identifica 5 componentes que permitan articular dicha arquitectura, considerando que toda aplicación de un sistema de información está caracterizada por siguiente:

- Presentación/Captación de la información.
- Procesos.
- Almacenamiento de la información.
- Puestos de trabajo
- Comunicaciones.

## Ventajas

- Centralización del control: los accesos, recursos y la integridad de los datos son controlados por el servidor de forma que un programa cliente defectuoso o no autorizado no pueda dañar el sistema. Esta centralización también facilita la tarea de poner al día datos u otros recursos (mejor que en las redes P2P)..
- Escalabilidad: se puede aumentar la capacidad de clientes y servidores por separado. Cualquier elemento puede ser aumentado (o mejorado) en cualquier momento, o se pueden añadir nuevos nodos a la red (clientes y/o servidores).
- Fácil mantenimiento: al estar distribuidas las funciones y responsabilidades entre varios ordenadores independientes, es posible reemplazar, reparar, actualizar, o incluso trasladar un servidor, mientras que sus clientes no se verán afectados por ese cambio (o se afectarán mínimamente). Esta independencia de los cambios también se conoce como encapsulación.

- Existen tecnologías, suficientemente desarrolladas, diseñadas para el paradigma de C/S que aseguran la seguridad en las transacciones, la amigabilidad de la interfaz, y la facilidad de empleo.

### Desventajas

- La congestión del tráfico ha sido siempre un problema en el paradigma de C/S. Cuando una gran cantidad de clientes envían peticiones simultáneas al mismo servidor, puede ser que cause muchos problemas para éste (a mayor número de clientes, más problemas para el servidor). Al contrario, en las redes P2P como cada nodo en la red hace también de servidor, cuanto más nodos hay, mejor es el ancho de banda que se tiene.
- El paradigma de C/S clásico no tiene la robustez de una red P2P. Cuando un servidor está caído, las peticiones de los clientes no pueden ser satisfechas. En la mayor parte de redes P2P, los recursos están generalmente distribuidos en varios nodos de la red. Aunque algunos salgan o abandonen la descarga; otros pueden todavía acabar de descargar consiguiendo datos del resto de los nodos en la red.
- El software y el hardware de un servidor son generalmente muy determinantes. Un hardware regular de un ordenador personal puede no poder servir a cierta cantidad de clientes. Normalmente se necesita software y hardware específico, sobre todo en el lado del servidor, para satisfacer el trabajo. Por supuesto, esto aumentará el coste.
- El cliente no dispone de los recursos que puedan existir en el servidor. Por ejemplo, si la aplicación es una Web, no podemos escribir en el disco duro del cliente o imprimir directamente sobre las impresoras sin sacar antes la ventana previa de impresión de los navegadores.

## Arquitectura en tres capas

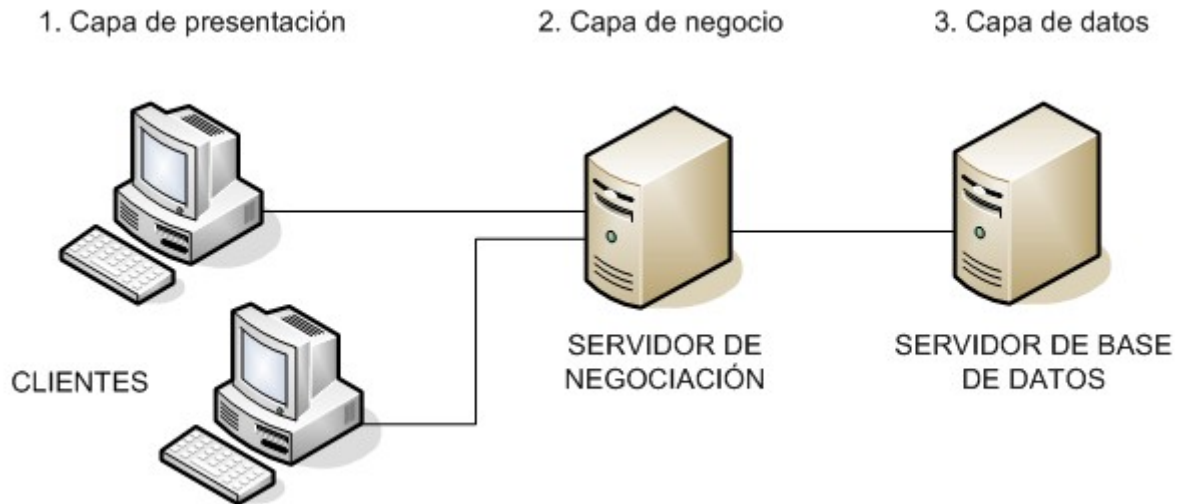
La Arquitectura multicapa o multinivel es un modelo de desarrollo software en el que el objetivo primordial es la separación (desacoplamiento) de las partes que componen un sistema software o también una arquitectura cliente-servidor. De esta forma, por ejemplo, es sencillo y mantenible crear diferentes interfaces sobre un mismo sistema sin requerirse cambio alguno en la capa de datos o lógica.

La ventaja principal de este estilo es que el desarrollo se puede llevar a cabo en varios niveles y, en caso de que sobrevenga algún cambio, solo afectará al nivel requerido sin tener que revisar entre el código fuente de otros módulos, dado que se habrá reducido el Acoplamiento informático hasta una interfaz de paso de mensajes.

Además, permite distribuir el trabajo de creación de una aplicación por niveles; de este modo, cada grupo de trabajo está totalmente abstraído del resto de niveles, de forma que basta con conocer la API que existe entre niveles.

En el diseño de sistemas informáticos actual se suelen usar las arquitecturas multinivel o programación por capas. En dichas arquitecturas a cada nivel se le confía una misión simple, lo que permite el diseño de arquitecturas escalables (que pueden ampliarse con facilidad en caso de que las necesidades aumenten).

El más utilizado actualmente es el diseño en tres capas.



## Capas y niveles

**Capa de presentación:** Es la que ve el usuario (también se la denomina «capa de usuario»), presenta el sistema al usuario, le comunica la información y captura la información del usuario en un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). También es conocida como interfaz gráfica y debe tener la característica de ser «amigable» (entendible y fácil de usar) para el usuario. Esta capa se comunica únicamente con la capa de negocio.

**Capa de negocio:** es donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio) porque es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos almacenar o recuperar datos de él. También se consideran aquí los programas de aplicación.

**Capa de datos:** es donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

## Instalación de un servidor

Breve resumen de los pasos para descargar e instalar Ubuntu Server en VirtualBox:

### Virtualbox

Según la empresa propietaria: *“Oracle VM VirtualBox, el software de virtualización multiplataforma de código abierto más popular del mundo, permite a los desarrolladores entregar código más rápido, ya que pueden ejecutar múltiples sistemas operativos en un solo dispositivo”*.

Se puede descargar de forma gratuita desde <https://www.virtualbox.org/>



Virtualbox te permite realizar tareas de virtualización de sistemas, con las ventajas de:

- Realizar pruebas de aplicaciones y sistemas operativos sin la necesidad de que se ejecuten en un ambiente real.
- Validar compatibilidad y rendimiento de programas en un sistema operativo específico.
- Evitar la presencia de malware o virus en las máquinas productivas.

Ya que se encarga de aislar el sistema operativo inicial del que vas a usar como invitado.

Entre las desventajas de la virtualización, el rendimiento de las máquinas virtuales es un poco más bajo que el de un servidor físico. La máquina física sobre la que se ejecutan las máquinas virtuales es muy crítica: Cualquier falla en uno de sus componentes hardware afectaría a todas las máquinas virtuales que usen este recurso.

### Configurar red interna

Para conectarte a un servidor Ubuntu instalado en un VirtualBox desde la misma máquina, puedes utilizar una conexión de red interna o "host-only" para establecer una conexión entre el host (tu máquina) y el invitado (la máquina virtual con Ubuntu).



Antes de crear la máquina virtual, se debe ir a Archivo > Administrador de red de anfitrión y crear una nueva red. Por defecto recibirá el nombre "vboxnet0".

## Instalar UBUNTU

1. Descarga la imagen de Ubuntu Server: Lo primero que debes hacer es descargar la imagen ISO de Ubuntu Server desde la página web oficial. Normalmente se sugiere la versión LTS (Long Term Support) más reciente de arquitectura 64 bits.
2. Crea una nueva máquina virtual: Abre VirtualBox y haz clic en "Nuevo" para crear una nueva máquina virtual. Asigna un nombre a la máquina y selecciona "Linux" como tipo y "Ubuntu (64-bit)" como versión.
3. Asigna la memoria RAM: Asigna la cantidad de memoria RAM que deseas utilizar para la máquina virtual. Por defecto virtualbox asigna 1 GB.
4. Crea un disco duro virtual: Selecciona "Crear disco duro virtual" y sigue las instrucciones para crear un nuevo disco duro virtual de al menos 10 GB.
5. Configura la máquina virtual: Selecciona la máquina virtual recién creada y haz clic en "Configuración". En la sección "Almacenamiento", selecciona "Agregar" y elige la imagen ISO de Ubuntu Server que descargaste en el primer paso.
6. Configura las conexiones de red, en la opción RED deja el primer adaptador en modo NAT para poder conectar a internet utilizando la conexión del anfitrión, y agrega un segundo adaptador de red en modo Adaptador solo anfitrión y seleccionando la red virtual creada anteriormente.
7. Inicia la máquina virtual: Haz clic en "Iniciar" para iniciar la máquina virtual y comenzar la instalación de Ubuntu Server.
8. Instala Ubuntu Server: Sigue las instrucciones en pantalla para instalar Ubuntu Server en la máquina virtual. Asegúrate de seleccionar la configuración de red adecuada y de instalar las actualizaciones de seguridad necesarias.

Algunos de los comandos básicos que debes conocer para utilizar Ubuntu:

- **ls:** Este comando te muestra el contenido de un directorio. Por defecto, muestra el contenido del directorio actual.
- **cd:** Este comando te permite cambiar de directorio. Por ejemplo, `cd Desktop` te llevará al directorio "Desktop".
- **mkdir:** Este comando te permite crear un nuevo directorio. Por ejemplo, `mkdir Documentos` creará un directorio llamado "Documentos".
- **rm:** Este comando te permite eliminar archivos o directorios. Por ejemplo, `rm archivo.txt` eliminará el archivo llamado "archivo.txt".



- **cp:** Este comando te permite copiar archivos o directorios. Por ejemplo, `cp archivo.txt carpeta/` copiará el archivo "archivo.txt" dentro de la carpeta "carpeta".
- **mv:** Este comando te permite mover o renombrar archivos o directorios. Por ejemplo, `mv archivo.txt carpeta/` moverá el archivo "archivo.txt" dentro de la carpeta "carpeta".
- **sudo:** Este comando te permite ejecutar comandos con privilegios de superusuario. Por ejemplo, `sudo apt-get update` actualizará el sistema.
- **apt-get:** Este comando te permite instalar o desinstalar paquetes en Ubuntu. Por ejemplo, `sudo apt-get install nombre-del-paquete` instalará el paquete especificado.
- **shutdown:** Este comando sirve para enviar el aviso de apagado del servidor a todos los anfitriones y luego de un tiempo (generalmente 1 minuto) se cierra el sistema y si es posible se apaga el servidor. Para apagar inmediatamente se debe utilizar el modificador `now`, por ejemplo `sudo shutdown now`, también se puede utilizar para hacer un reset mediante la orden `sudo shutdown -r now`.

Otro comando muy importante es **ifconfig**, se utiliza para mostrar y configurar las interfaces de red en un sistema.

Con `ifconfig`, puedes ver información detallada sobre las interfaces de red disponibles en el sistema, como su dirección IP, máscara de subred, dirección MAC y otras configuraciones de red. También puedes utilizar `ifconfig` para configurar o modificar la configuración de una interfaz de red, como cambiar su dirección IP o activar o desactivar la interfaz.

### ***Algunos ejemplos de uso comunes de ifconfig:***

Mostrar información detallada sobre todas las interfaces de red disponibles en el sistema:

```
ifconfig -a
```

Mostrar información detallada sobre una interfaz de red específica, como `eth0`:

```
ifconfig eth0
```

Configurar la dirección IP de una interfaz de red, como `eth0`, a `192.168.1.100`:

```
sudo ifconfig eth0 192.168.1.100
```

Activar o desactivar una interfaz de red, como `eth0`:

```
sudo ifconfig eth0 up # para activar la interfaz
```

```
sudo ifconfig eth0 down # para desactivar la interfaz
```

## Usar IP en lugar de ifconfig

En las últimas versiones de Ubuntu Server, el comando "ifconfig" ha sido reemplazado por el comando "ip" para configurar las interfaces de red. El comando "ip" es más completo y flexible que "ifconfig", y proporciona una forma más moderna de administrar la configuración de red.

Para mostrar la información de la interfaz de red, puedes utilizar el siguiente comando:

```
ip address show
```

Este comando muestra la información detallada de todas las interfaces de red en el sistema, incluyendo sus direcciones IP, máscaras de subred y otras configuraciones de red.

Para configurar la dirección IP de una interfaz de red, puedes utilizar el siguiente comando:

```
sudo ip address add <dirección IP>/<máscara de subred> dev <nombre de interfaz>
```

Reemplaza "<dirección IP>" y "<máscara de subred>" con la dirección IP y máscara de subred que deseas configurar, y "<nombre de interfaz>" con el nombre de la interfaz de red que deseas configurar (por ejemplo, eth0 o enp0s3).

Por ejemplo, para configurar la dirección IP de la interfaz eth0 a 192.168.1.100 con una máscara de subred de 255.255.255.0, puedes utilizar el siguiente comando:

```
sudo ip address add 192.168.1.100/24 dev eth0
```

Para activar o desactivar una interfaz de red, puedes utilizar el siguiente comando:

```
sudo ip link set <nombre de interfaz> up # para activar la interfaz
```

```
sudo ip link set <nombre de interfaz> down # para desactivar la interfaz
```

Reemplaza "<nombre de interfaz>" con el nombre de la interfaz que deseas activar o desactivar (por ejemplo, eth0 o enp0s3).

Estos son solo algunos de los comandos básicos que debes conocer para utilizar Ubuntu. Hay muchos otros comandos y opciones disponibles, pero estos te darán una buena base para comenzar a trabajar con la línea de comandos de Ubuntu.

## Conectar desde el sistema anfitrión

Por lo general al instalar ubuntu, si los pasos se realizaron correctamente la instalación detectará ambos adaptadores de red y tendremos dos direcciones IP: la de internet y la de la red virtual.

Utiliza el comando "ip address show" para determinar la dirección IP asignada a la interfaz de red "Adaptador solo anfitrión". La dirección IP debería estar en el rango de direcciones IP de la red "host-only" configurada en VirtualBox.

Desde la terminal en el host, utiliza el comando "ssh" para conectarte al servidor Ubuntu en la máquina virtual. El comando ssh se utiliza para establecer una conexión SSH segura con el servidor Ubuntu. Debes utilizar la dirección IP asignada a la interfaz "Adaptador solo anfitrión" como la dirección del servidor.

Por ejemplo, si la dirección IP de la interfaz "Adaptador solo anfitrión" en la máquina virtual es "192.168.56.101", puedes conectarte al servidor Ubuntu utilizando el siguiente comando desde la terminal en el host:

```
ssh usuario@192.168.56.101
```

Reemplaza "usuario" con el nombre de usuario que utilizaste para crear la cuenta en Ubuntu durante la instalación.

Al ejecutar este comando, se iniciará una conexión SSH segura con el servidor Ubuntu en la máquina virtual, y podrás utilizar la terminal en el host para interactuar con el servidor.

Ten en cuenta que es posible que debas instalar un cliente SSH en el anfitrión con Windows, ya que no viene incluido por defecto en el sistema operativo. Puedes utilizar el cliente OpenSSH incluido en la suite de herramientas de Cygwin o instalar un cliente SSH independiente como PuTTY.

## FTP

FTP significa "File Transfer Protocol" (Protocolo de Transferencia de Archivos en español). Es un protocolo de red utilizado para transferir archivos de un servidor a un cliente a través de Internet.

El FTP fue creado en la década de 1970 y se ha convertido en uno de los protocolos más comunes para la transferencia de archivos en Internet. A través de una conexión FTP, los usuarios pueden acceder a archivos almacenados en un servidor remoto, descargarlos o subir nuevos archivos al servidor.

FTP se basa en un modelo cliente-servidor, donde el cliente se conecta al servidor FTP para realizar operaciones de transferencia de archivos. Por lo general, se utiliza un software cliente de FTP, como FileZilla o WinSCP, para conectarse a un servidor FTP. El cliente de FTP solicita al servidor la transferencia de un archivo y este último envía los datos correspondientes al cliente.

FTP es un protocolo relativamente simple y fácil de usar, pero no es tan seguro como otros protocolos de transferencia de archivos más recientes, como SFTP (SSH File Transfer Protocol) o FTPS (FTP seguro). Sin embargo, aún se utiliza ampliamente en la transferencia de archivos en línea.

Para instalar un servidor FTP en Ubuntu, puedes seguir los siguientes pasos:

Abre una terminal y actualiza la lista de paquetes con el comando:

```
sudo apt-get update
```

Instala el servidor FTP vsftpd con el comando:

```
sudo apt-get install vsftpd
```

Una vez instalado, abre el archivo de configuración de vsftpd con el comando:

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

Configura el servidor FTP según tus necesidades. Puedes editar el archivo vsftpd.conf para establecer la configuración que desees. Por ejemplo, puedes especificar el directorio raíz para los archivos FTP, la cantidad máxima de conexiones simultáneas permitidas, y otras opciones.

Reinicia el servicio FTP con el comando:

```
sudo systemctl restart vsftpd
```

Con estos pasos, habrás instalado y configurado un servidor FTP en Ubuntu. Para acceder al servidor FTP, puedes utilizar un cliente FTP como FileZilla o cualquier otro software de FTP. Para conectarte al servidor FTP, necesitarás conocer la dirección IP del servidor, el puerto de FTP y las credenciales de inicio de sesión.

## SSH

SSH significa "Secure Shell" (Shell Seguro en español). Es un protocolo de red que permite a los usuarios conectarse y comunicarse de forma segura con dispositivos remotos a través de Internet.

SSH se utiliza para establecer una conexión cifrada y segura entre dos dispositivos. Esto permite a los usuarios conectarse a un dispositivo remoto (como un servidor) y ejecutar comandos en ese dispositivo de forma segura. SSH utiliza criptografía para proteger la conexión y evitar que terceros puedan interceptar o manipular los datos que se transfieren a través de la conexión.

Además de permitir la conexión segura a dispositivos remotos, SSH también se utiliza para transferir archivos de forma segura mediante SFTP (SSH File Transfer Protocol). SFTP es similar a FTP (File Transfer Protocol), pero utiliza SSH para proteger la transferencia de archivos.

SSH es ampliamente utilizado en la administración de sistemas y en el desarrollo de software. Los administradores de sistemas utilizan SSH para administrar servidores remotos, mientras que los desarrolladores utilizan SSH para conectarse a servidores de prueba o producción y realizar tareas de desarrollo o de mantenimiento.

En general la versión SERVER de ubuntu ofrece habilitar un servidor SSH en la instalación, si no se ha aceptado esa opción, para instalar un servidor SSH en Ubuntu, puedes seguir los siguientes pasos:

Abre una terminal y actualiza la lista de paquetes con el comando:

```
sudo apt-get update
```

Instala el servidor SSH OpenSSH Server con el comando:

```
sudo apt-get install openssh-server
```

Una vez instalado, el servicio de SSH se iniciará automáticamente. Puedes verificar que el servicio está en ejecución con el siguiente comando:

```
sudo systemctl status ssh
```

Si deseas modificar la configuración de SSH, puedes hacerlo editando el archivo de configuración `/etc/ssh/sshd_config` utilizando tu editor de texto preferido.

Para conectarte al servidor SSH, necesitarás conocer la dirección IP del servidor. Si estás en la misma red, puedes utilizar la dirección IP local del servidor. Si estás conectado a través de Internet, necesitarás la dirección IP pública del servidor.

Con estos pasos, habrás instalado y configurado un servidor SSH en Ubuntu. Para conectarte al servidor SSH, puedes utilizar un cliente SSH como PuTTY (para Windows) o el cliente SSH integrado en la línea de comandos de Ubuntu (para Linux y macOS).

## TheSSS

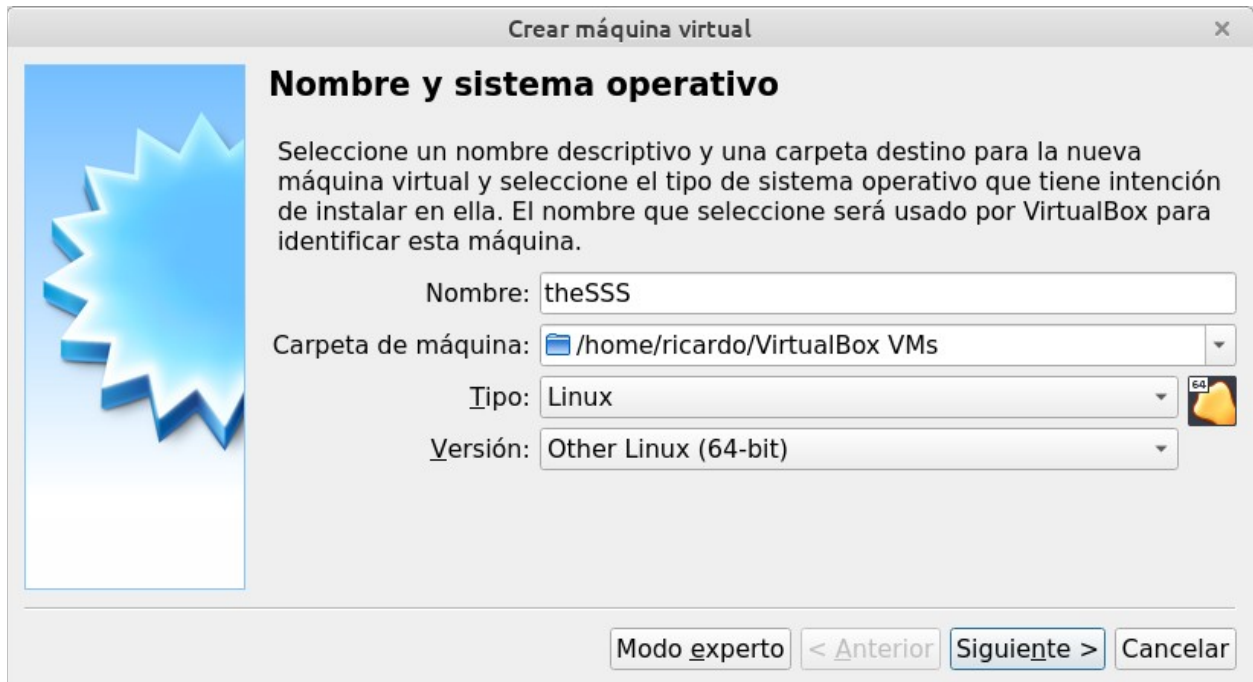
Vamos a practicar con la instalación de otro sistema operativo, en este caso será The **S**mallest **S**erver **S**uite (**TheSSS**), una distribución ligera de servidor Linux basado en **4MLinux**, la cual es una distribución de Linux en miniatura de origen polaco con más de una década de trayectoria a sus espaldas que no esta basada directamente en ninguna otra distribución. Está especialmente diseñado para administradores de sistemas que desean configurar un conjunto simple y pequeño de software de servidor para pequeñas redes de área local. También puede usarse para desarrolladores que quieran probar sus aplicaciones sin instalar una configuración compleja de servidor Linux. Como su nombre lo indica, el tamaño de esta pequeña distribución de Linux es de menos de 170 MB.

Usando TheSSS, podemos instalar y configurar FTP, HTTP, SFTP, SSH, servidor Proxy y Telnet en muy poco tiempo. TheSSS viene con 4MLinux Firewall y Clam Antivirus por defecto. Puede crear un USB en vivo y probarlo para verificar si se ajusta a sus necesidades, o instalarlo en cualquier máquina virtual y comenzar a explorarlo de inmediato.

Para descargar la versión SERVER lo puede hacer desde el sitio <https://sourceforge.net/projects/thesss/>. La página del proyecto está en <https://thesss.4mlinux.com/>

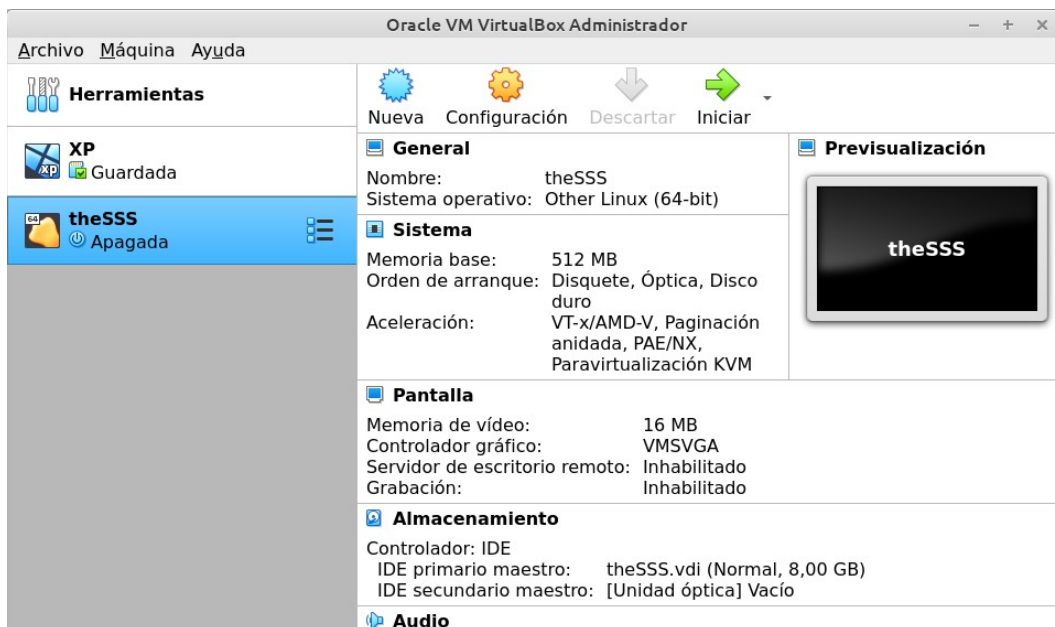
## Preparación

Ahora hay que configurar una máquina virtual en VirtualBox:

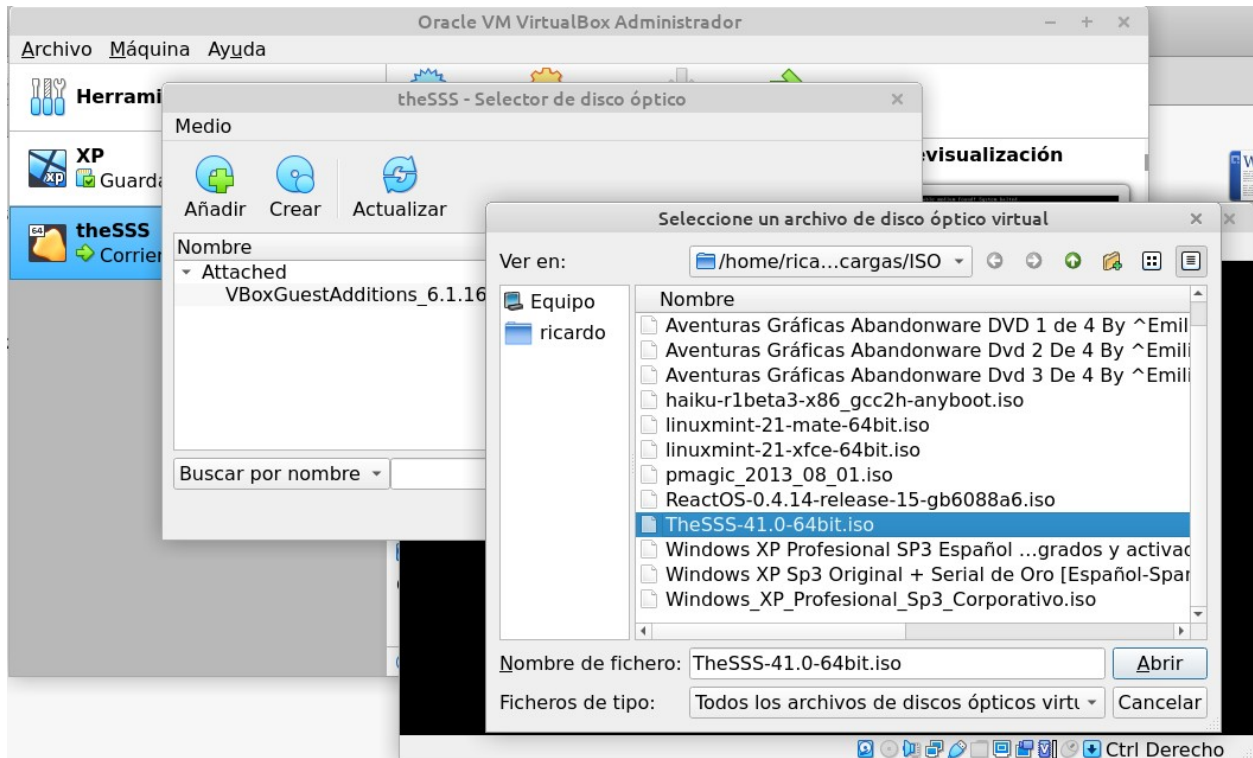


Las características de la máquina virtual van a ser 512MB RAM (en la página de la distribución se recomiendan 256) y 8 GB de disco (en la página hablan de alrededor de 5GB), si usted piensa que es insuficiente puede agrandar ambos valores.

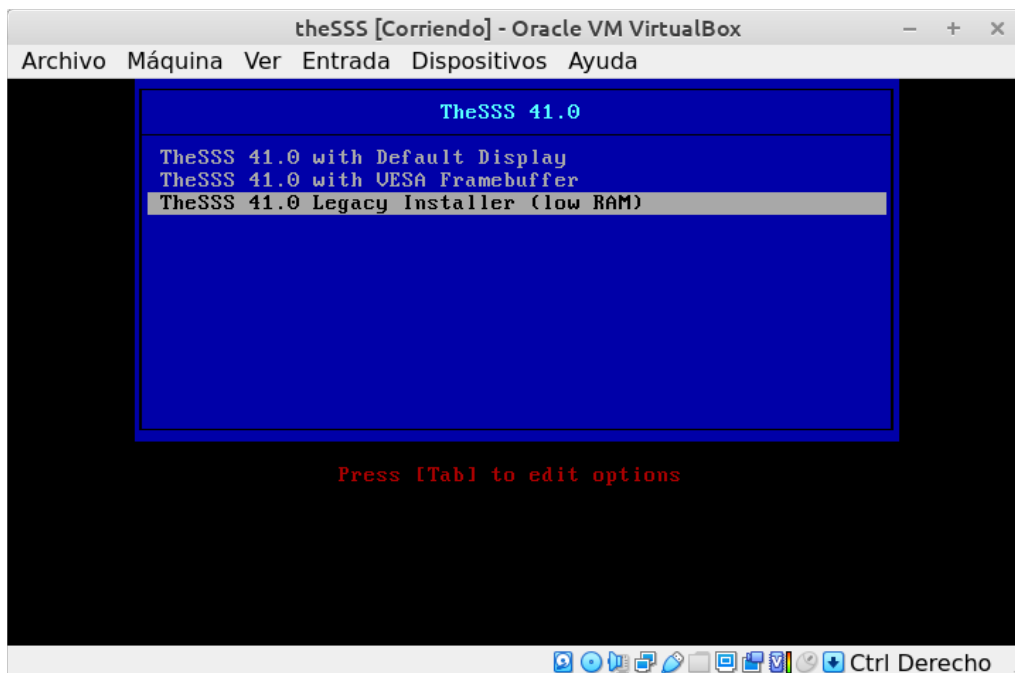
Recuerde también agregar el segundo adaptador de red para la red sólo anfitrión configurada la clase anterior.



Al momento de arrancar la máquina virtual se nos solicita que añadamos un soporte, agregamos la ISO descargada.

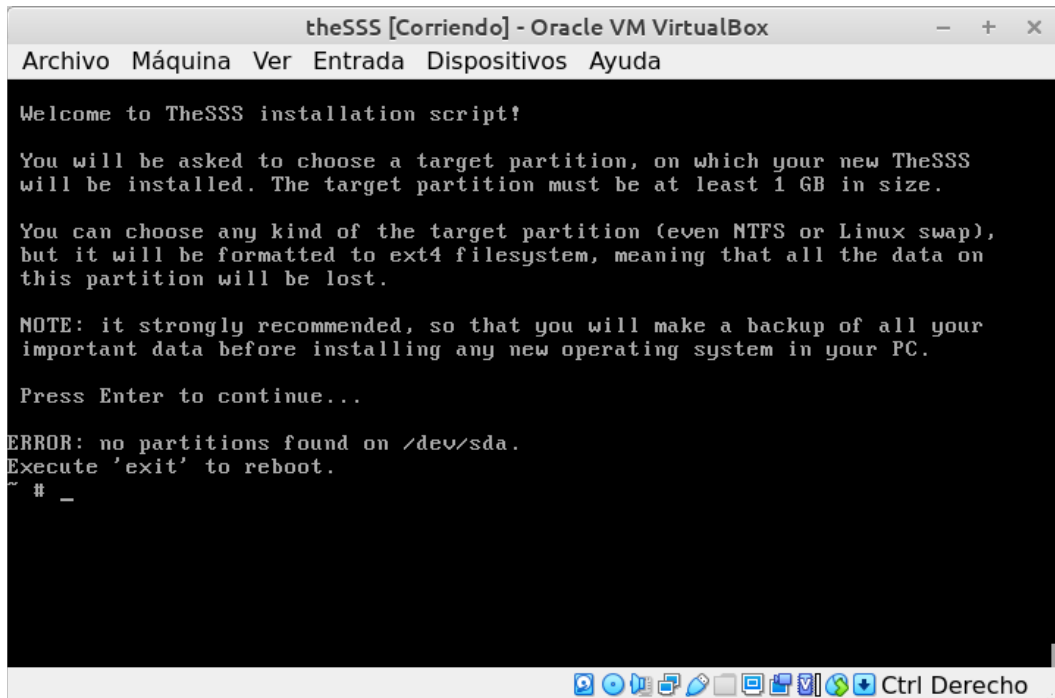


Cuando arranca nos mostrará una imagen similar a la siguiente, como hemos configurado poca RAM, vamos a elegir la opción Legacy Installer.



Al aceptar se ejecutará automáticamente el script de instalación, pidiendo que presionemos Enter para continuar, pero este fallará porque no encuentra una partición para instalarse ya que el disco que agregamos cuando creamos la máquina virtual está sin particionar.





theSSS [Corriendo] - Oracle VM VirtualBox

Archivo Máquina Ver Entrada Dispositivos Ayuda

```
Welcome to TheSSS installation script!

You will be asked to choose a target partition, on which your new TheSSS
will be installed. The target partition must be at least 1 GB in size.

You can choose any kind of the target partition (even NTFS or Linux swap),
but it will be formatted to ext4 filesystem, meaning that all the data on
this partition will be lost.

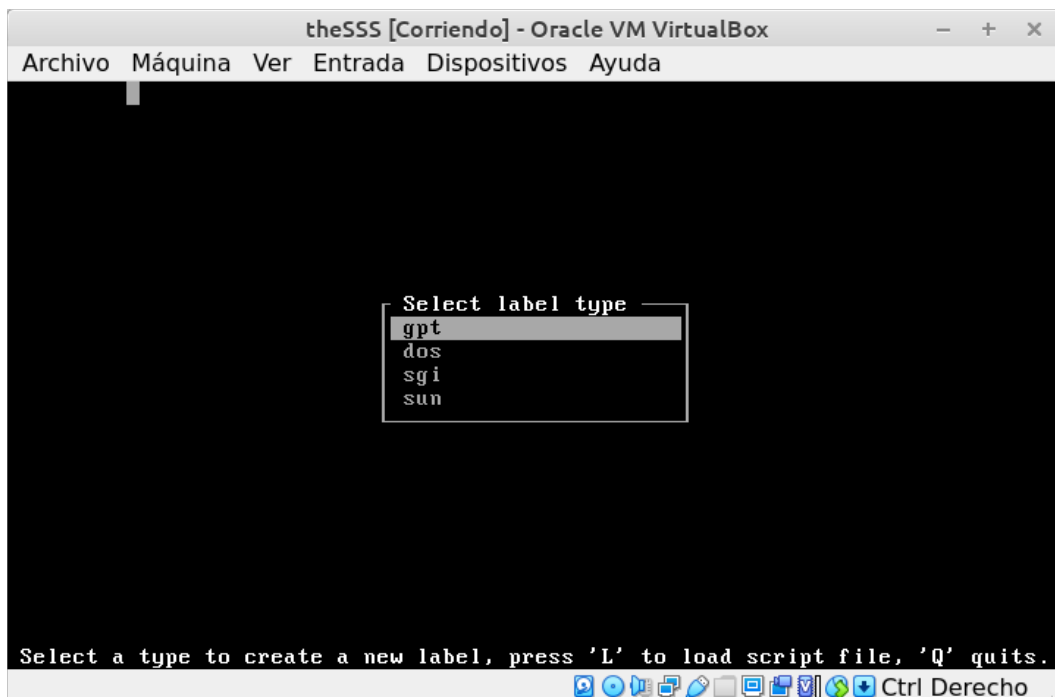
NOTE: it strongly recommended, so that you will make a backup of all your
important data before installing any new operating system in your PC.

Press Enter to continue...

ERROR: no partitions found on /dev/sda.
Execute 'exit' to reboot.
~ # _
```

Ctrl Derecho

Para particionar el disco podemos utilizar el comando **cfdisk**. Al ejecutarlo nos pide que seleccionemos la estructura de la tabla de particiones, elegimos el tipo GPT que es preferible para los BIOS UEFI, tengamos en cuenta que la opción DOS será para seleccionar una tabla de tipo MBR aceptadas para BIOS de máquinas antiguas.



theSSS [Corriendo] - Oracle VM VirtualBox

Archivo Máquina Ver Entrada Dispositivos Ayuda

```
Select label type
gpt
dos
sgi
sun
```

Select a type to create a new label, press 'L' to load script file, 'Q' quits.

Ctrl Derecho

Vamos a crear dos particiones.

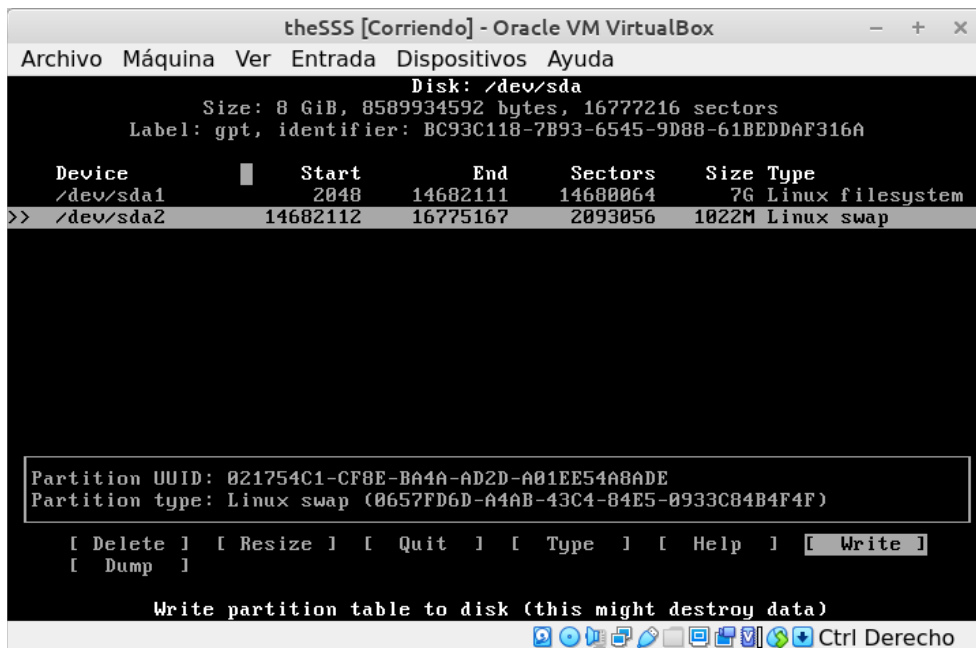
- Partición raíz, /dev/sda1, como la principal con un tamaño de 7 GB y formato ext4.

- Partición de intercambio (swap), /dev/sda2, con tamaño restante de aproximadamente 1024 MB (se calcula como el doble de RAM).

Se pueden crear tantas particiones como se desee, como por ejemplo una partición HOME separada, pero para esta práctica vamos a crear sólo estas dos particiones.

Seleccione '**Nuevo**'. Ingrese el tamaño de la partición, en nuestro caso es de 7G donde la G simboliza GB. También puede especificar el tamaño en MB usando la letra 'M'. Con esto se hará una partición de tipo Linux Filesystem.

Luego seleccionar otra vez Nuevo, hacer una partición con los 1023 restantes y luego cambiar el tipo de partición como "**Linux Swap**".



Presionar la opción "**Write**" y luego "**yes**" para confirmar la escritura de la tabla de particiones. Para finalizar salir con la opción "**Quit**".

Luego, escriba **udev** en el indicador para guardar la tabla de particiones. **Udev** es el administrador de dispositivos para el kernel **de Linux** que crea/elimina dinámicamente nodos de dispositivos en el directorio /dev.

udev

Puede verificar los detalles de la partición usando el comando:

fdisk -l

```

theSSS [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda

Disk /dev/ram14: 16 MiB, 16777216 bytes, 32768 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes

Disk /dev/ram15: 16 MiB, 16777216 bytes, 32768 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes

Disk /dev/sda: 8 GiB, 8589934592 bytes, 16777216 sectors
Disk model: VBOX HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: BC93C118-7B93-6545-9D88-61BEDDAF316A

Device      Start      End  Sectors  Size Type
/dev/sda1    2048 14682111 14680064    7G Linux filesystem
/dev/sda2  14682112 16775167  2093056 1022M Linux swap
~ #

```

Si bien el instalador lo volverá a hacer, para practicar podemos practicar con formatear la partición del sistema y activar las partición de swap:

Formatee la partición primaria usando el comando:

```
mkfs.ext4 /dev/sda1
```

Luego, formatee y active la partición de intercambio usando los siguientes comandos:

```
mkswap /dev/sda2
swapon /dev/sda2
```

## Instalación del sistema

Es hora de instalar TheSSS en nuestro sistema. Para hacerlo, escriba

```
install2hd
```

```

theSSS [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda

Welcome to TheSSS installation script!

You will be asked to choose a target partition, on which your new TheSSS
will be installed. The target partition must be at least 1 GB in size.

You can choose any kind of the target partition (even NTFS or Linux swap),
but it will be formatted to ext4 filesystem, meaning that all the data on
this partition will be lost.

NOTE: it strongly recommended, so that you will make a backup of all your
important data before installing any new operating system in your PC.

Press Enter to continue..._

```

A continuación, ingrese el identificador de la partición de destino para instalar el servidor TheSSS. En este caso será *sda1*.

Luego nos dirá que se va a formatear con el sistema de archivos ext4, y consultará si queremos cambiar, la opción que elegiremos será decirle que no, pero si usted quiere cambiarlo le ofrecerá otras opciones como ext2, ext3, reiserfs, ntfs y otros tipos de sistemas de archivos.

```

theSSS [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda

Disk /dev/sd: 8 GiB, 8589934592 bytes, 16777216 sectors
Disk model: VBox HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: BC93C118-7B93-6545-9D88-61BEDDAF316A

Device      Start      End  Sectors  Size Type
/dev/sda1    2048 14682111 14680064   7G Linux filesystem
/dev/sda2   14682112 16775167 2093056 1022M Linux swap

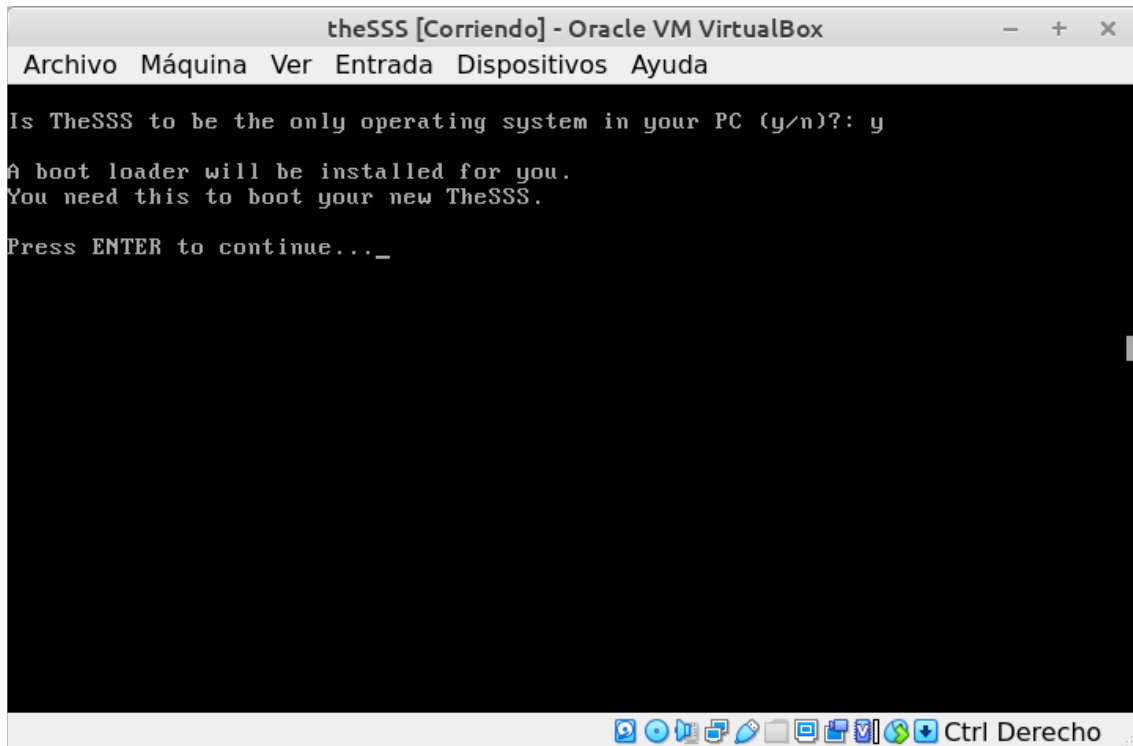
On which partition should your new TheSSS be installed?
Give the name (e.g., sda1) of the target partition: sda1

/dev/sda1 will be formatted to ext4.
Do you wish to change this (y/n)?:_

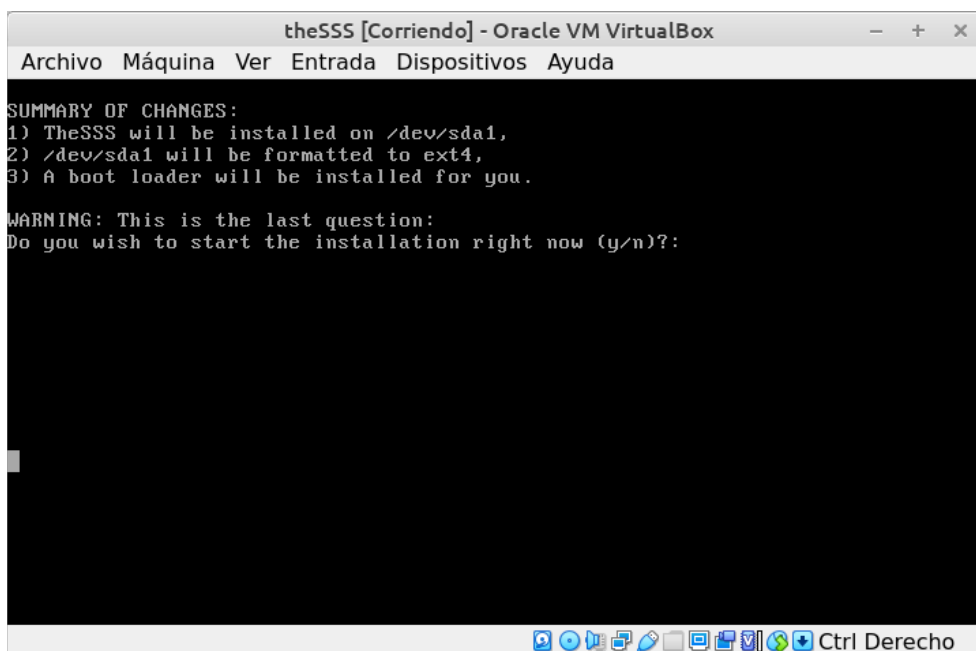
```

La última pregunta es si este será el único sistema operativo que podrá arrancar esta PC, en este caso diremos que si. Si le decimos que NO supondrá que podemos llegar a instalar otro sistema operativo e instalará un administrador de arranque como el GRUB.

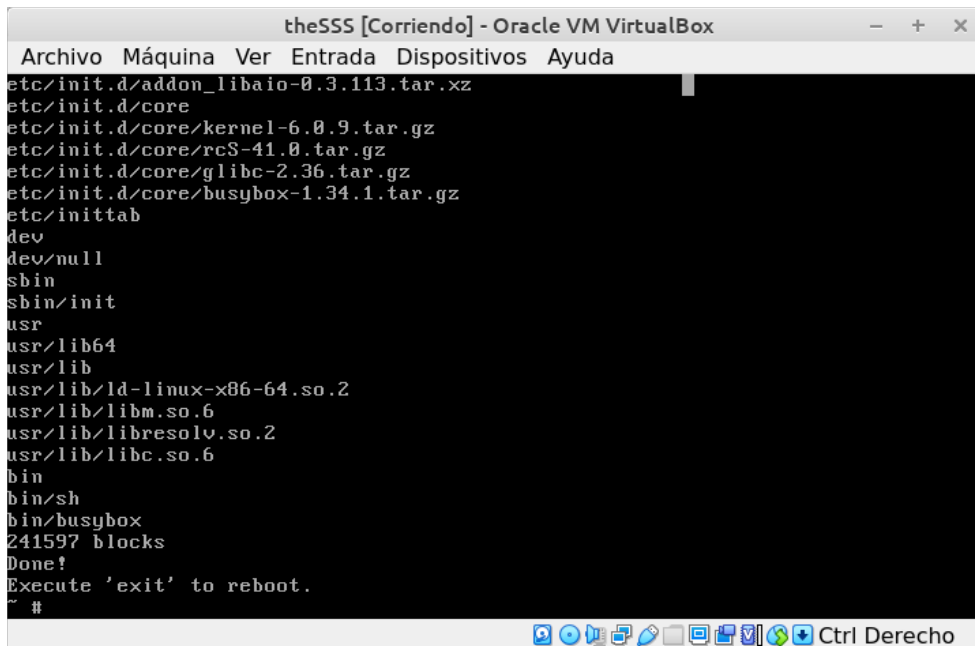
Al aceptar nos indicará que se instalará un boot loader en este caso uno muy simple llamado LILO.



A continuación, verá el resumen de cambios hasta el momento. Si todo está bien, escriba 'y' para iniciar la instalación.



Todos los archivos de instalación se copiarán en la partición de destino. Una vez que se complete el proceso de copia, volverá al indicador de bash.



Ahora deberá reiniciar la máquina para que arranque desde la instalación. Primero vaya al menú de la ventana de virtualbox en Dispositivos → unidades ópticas → eliminar disco de la unidad virtual (a lo mejor deberá forzar a que lo haga). Luego deberá ejecutar la orden

```
exit
```

## Configuración de servicios

Una vez reiniciada la máquina, veremos que instala un montón de módulos, entre ellos MariaDB y PHP 5 y 7. Luego nos pedirá que seleccionemos la distribución de teclado que poseemos, no hay demasiadas alternativas

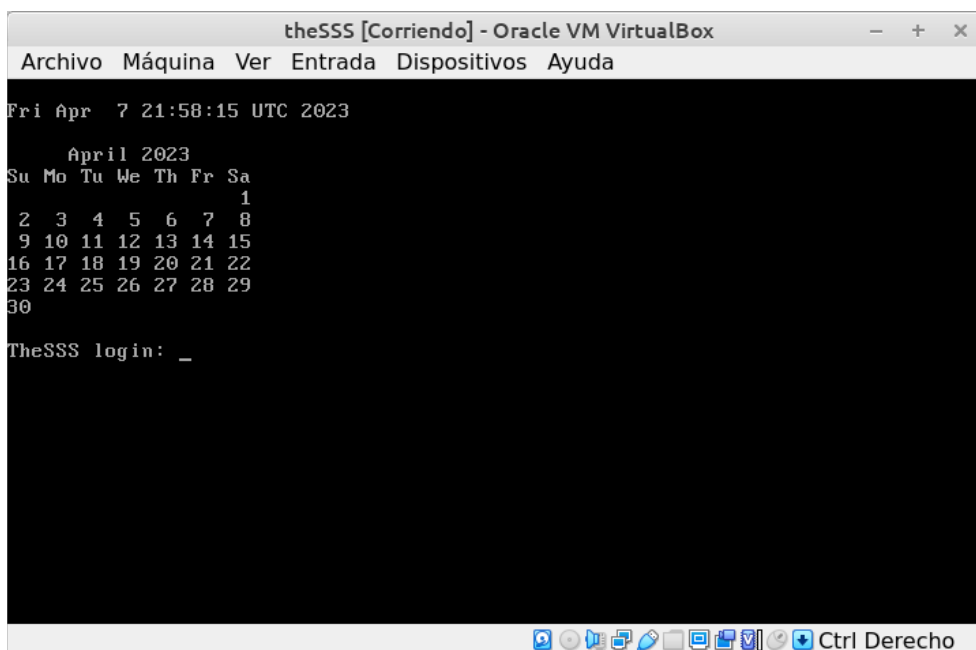


A continuación se le pedirá que ingrese la nueva contraseña para el usuario root. Ingrese cualquier contraseña segura de su elección.

Luego, al ser esta una distribución polaca, nos preguntará si vivimos en europa, para configurar la zona horaria por defecto. Como no es el caso nos informa que deberemos hacerlo con la orden tzselect.

```
***LOCAL TIME***  
  
Do you live in Europe (y/n)? n  
  
Your location: not set (UTC)  
Your local time: 22:11  
  
Execute 'tzselect' in terminal to change these settings.  
Press Enter to continue._
```

Finalmente, verá el indicador de usuario raíz como se muestra a continuación. Introduzca la contraseña del usuario root.



Al ingresar nos comenta que en cualquier momento podemos ver la ayuda con el comando helpme.



```

theSSS [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda

Welcome to TheSSS (The Smallest Server Suite) !

The following three commands are the most important for you:
- server,
- netconfig,
- install2hd.
I hope that their names are self-explanatory :-)
```

ADDITIONAL APPLICATIONS:

```

Backup tools: backup, fsbackup, image, fsarchiver, rsync.
Set of system monitoring tools: nmonitor.
Script to install and run ClamAV: antivir.
Script to replace LILO with GRUB: update-grub.
FTP utilities: gftp, lftp, psftp, curlftpfs.
SSH utilities: ssh, plink, dbclient, sshfs.
IRC utilities: ircii, irssi.
NBD utilities: nbd-client, nbd-server.
Download utilities: curl, wget.
Network scanner: nmap.
Bittorrent client: rtorrent.
Disk partition managers: fdisk, cfdisk, parted.
Recovery tools: testdisk (lost partitions), photorec (lost files).
Utility to erase the entire contents of a disk: wipe.

BONUS PACK INCLUDED! Execute the 'bonus' command to learn more.

root@TheSSS:~#
```

Con el comando `server` veremos la lista de servicios (daemons) que se encuentran instalados.

```

theSSS [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda

List of server daemons:
dnssd --> BIND (standalone)
firewall --> iptables (4MLinux Firewall)
fspd --> FSP daemon (standalone)
ftpd --> ProFTPD (standalone)
httpd --> Apache (standalone)
mysqld --> MariaDB (standalone)
nfsd --> NFS daemon (standalone)
proxyd --> Polipo (standalone)
smtpd --> Postfix (standalone)
sshd --> OpenSSH (standalone)
telnetd --> BusyBox (via inetd)
tftpd --> BusyBox (via inetd)
serverd --> script to control all servers

Example of start/stop commands:
ftpd start --> start FTP server
ftpd stop --> stop FTP server
ftpd restart --> restart FTP server
ftpd test --> test FTP server

Execute 'php7' ('php5') to enable PHP7 (PHP5).
Execute 'webmin' to perform its initial setup.
The 4MLinux Server WebGUI: http://localhost/admin
root@TheSSS:~#
```

Pero no todo puede ser perfecto, si pedimos que nos muestre el estado de las interfaces de red, nos encontramos con que eth1 no se enteró que se tenía que conectar con la red sólo anfitrión.

```
root@TheSSS:~# ifconfig -a
eth0      Link encap:Ethernet  HWaddr 08:00:27:6D:F4:E6
          inet addr:10.0.2.15  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe6d:f4e6/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:8 errors:0 dropped:0 overruns:0 frame:0
          TX packets:17 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1743 (1.7 KiB)  TX bytes:1854 (1.8 KiB)

eth1      Link encap:Ethernet  HWaddr 08:00:27:75:97:DC
          BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

Para solucionar esto debemos ejecutar

```
udhcpc -i eth1
```

```
root@TheSSS:~# udhcpc -i eth1
udhcpc: started, v1.34.1
udhcpc: broadcasting discover
udhcpc: broadcasting select for 192.168.56.101, server 192.168.56.100
udhcpc: lease of 192.168.56.101 obtained from 192.168.56.100, lease time 600
root@TheSSS:~# ifconfig eth1
eth1      Link encap:Ethernet  HWaddr 08:00:27:75:97:DC
          inet addr:192.168.56.101  Bcast:192.168.56.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe75:97dc/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:2 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1180 (1.1 KiB)  TX bytes:1200 (1.1 KiB)

root@TheSSS:~#
```

Ahora vamos a testear algún servicio, por ejemplo podemos ir al navegador y tratamos de acceder al servicio http de la IP del linux virtualizado, nos debería salir una pantalla como esta:



## No se puede acceder a este sitio

**192.168.56.101** rechazó la conexión.

Intenta:

- Comprobar la conexión.
- [Comprobar el proxy y el firewall.](#)

ERR\_CONNECTION\_REFUSED

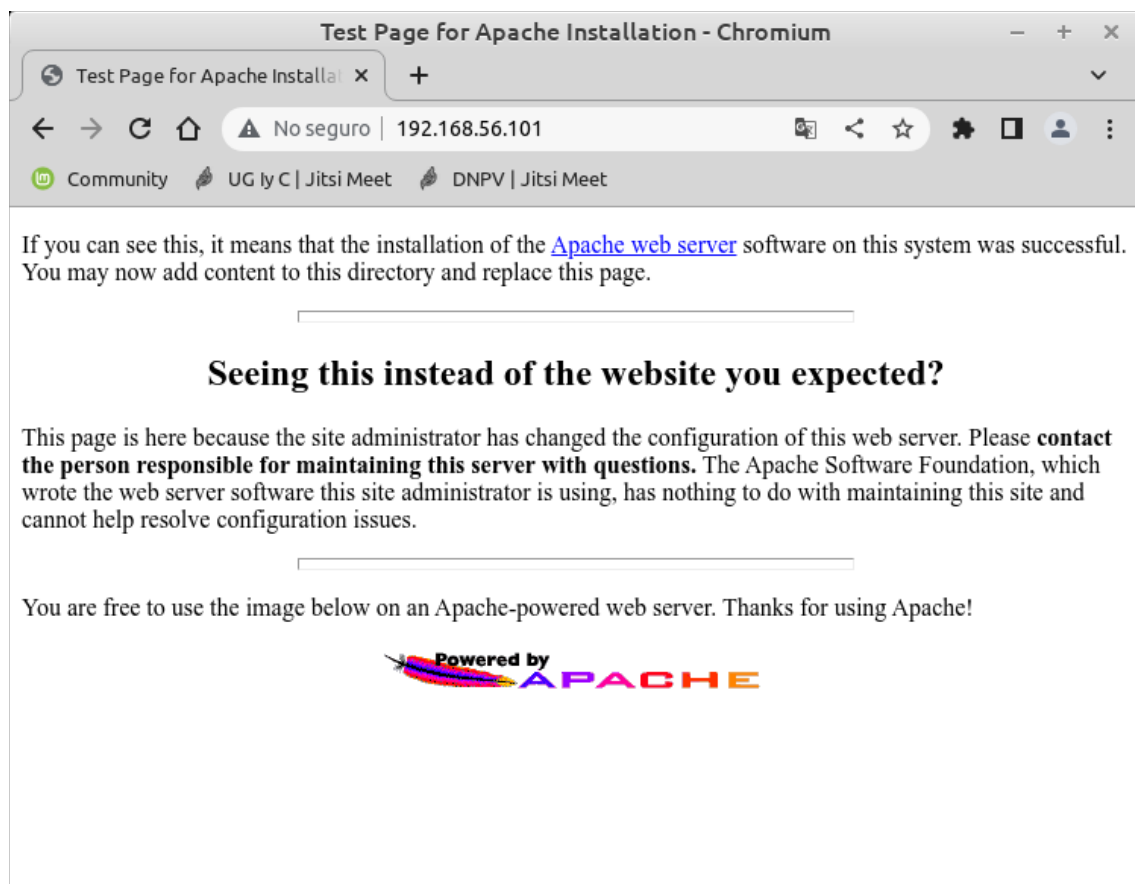
Entonces debemos iniciar el servicio httpd que vimos cuando ejecutamos el comando server.

Para iniciarlo debemos ejecutar el comando

```
httpd start
```

```
root@TheSSS:~# httpd
sed: unmatched '/'
Usage:
sed: unmatched '/'
httpd start --> start HTTP server
httpd stop --> stop HTTP server
httpd restart --> restart HTTP server
httpd test --> test HTTP server
root@TheSSS:~# httpd start
sed: unmatched '/'
root@TheSSS:~#
```

Y ahora recargamos la página en el navegador:



## Estableciendo la configuración de arranque

Ahora vamos a hacer otra prueba, reiniciemos el equipo con el comando:

```
restart
```

De paso comentemos que el comando para apagar la máquina es:

**poweroff**

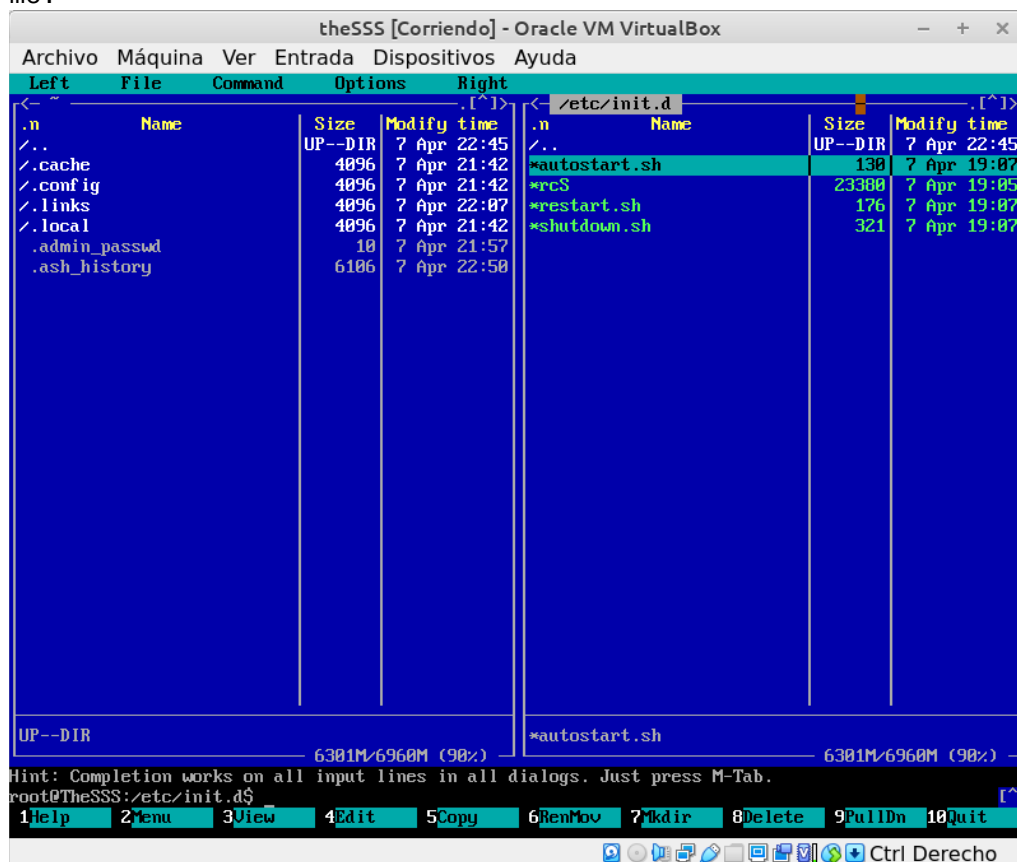
Cuando la máquina reinicia tratamos de conectarnos de nuevo con el navegador y vemos que la configuración de red y el server no guardaron la configuración. Así que para que la configuración sea permanente vamos a editar el archivo `/etc/init.d/autostart.sh`.

Una forma es usar el editor incorporado nano con el comando

```
nano /etc/init.d/autostart.sh
```

pero otra forma es utilizar el administrador de archivo midnight commander ejecutando el comando:

```
mc .
```



Nos dirigimos al directorio indicado y cuando se resalta el archivo `autostart.sh` se presiona F4 para ingresar al editor. Debemos agregar los comandos al final del script para que se ejecuten en cada arranque.

```
theSSS [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
autostart.sh [-M--] 0 L:[ 1+ 8 9/ 9] *(215 / 215b) <EOF> [*][X]
cupsd Z>/dev/null
modprobe loop Z>/dev/null
mount -a Z>/dev/null &
/usr/share/server/autostart.sh Z>/dev/null &
uvesa set 800x600
# comandos agregados para que se ejecuten automaticamente
udhcpd -i eth1
httpd start
```