Monta Tu Aula De Informática Con Raspberry Pi

- Introducción
- ¿Tiene sentido tener un aula de informática?
- Un poco de Historia
- 1. Materiales necesarios: Hardware y software
- 2. Instalación de Ubuntu 16
- 3. Instalación de PiNet
- 4. Instalación del sistema en la Raspberry Pi
- 5. Creación de usuarios. Creación de Carpetas Compartidas
- 6. Instalación de un nuevo programa: “ Epoptes”
- 7. Otras opciones del Menú.
- 8. Software educativo incluído en la Raspberry Pi
- Créditos

Copyright 2026

Introducción

Nadie duda hoy en día del papel fundamental de la tecnología en educación, siendo muchos los esfuerzos dedicados a normalizar y extender su uso en los centros educativos.

Hemos pasado por diferentes etapas en estos intentos: montaje de aulas de informática dedicadas, el programa *Escuela 2.0*... llegando a dotar a los centros con un portátil por alumno. En poco tiempo, el paradigma tecnológico ha cambiado tanto que las respuestas de las administraciones quedaban obsoletas poco después de haberse implementado.

El boom de los dispositivos móviles supuso de nuevo un paso de gigante en este camino, sembrando el terreno para que aflorara todo un ecosistema de *Apps* y soluciones basadas en la web. Esto ha hecho que poco a poco las prioridades hayan ido cambiando. Ahora hace falta una mejor conectividad a internet, una fibra potente, con redes preparadas para dar servicio a muchos más dispositivos. Sin embargo, no necesitamos comprarnos un gran ordenador o dispositivo móvil para que nos de un servicio aceptable.

En este entorno cambiante, las soluciones libres tanto en hardware como en software, ponen a nuestro alcance y de una manera sencilla todo lo que necesitamos para ser ciudadanos de nuestro tiempo.

https://giphy.com/embed/Ub3NWNmqNomU8

via GIPHY

{% include "git+https://github.com/catedu/robotica.git/README.md" %}

Copyright 2026

¿Tiene sentido tener un aula de informática?

Si tenemos dispositivos móviles y portátiles, ¿para qué queremos un aula de informática? Actualmente, escribo esto a día 21/12/2016, hay empresas como **Apple, Samsung y HP** que ofrecen programas muy ventajosos para modernizar nuestros colegios, con **asesoramiento y asistencia técnica** que libera al centro educativo de la labor de gestión de la red y los dispositivos, ya sean portátiles o tablets.

Una solución diferente es la que ofrece el **Departamento de Educación del Gobierno de Aragón con Vitalinux**. Podemos instalar Linux en varios ordenadores del centro o en todos, y ellos, a través de Migasfree, se encargan de la gestión remota de los equipos, manteniéndolos actualizados, instalando las aplicaciones que necesitemos y resolviendo incidencias. También simplifica enormemente el inventariado de todo nuestro equipamiento informático.

Si hablamos de **portátiles baratos**, sin duda la opción actual es **Pinebook**. Se trata un portátil basado en un procesador ARM y que cuenta con versiones de 11,6 (89 $) y 14 pulgadas (99 $). SoC Allwinner A64 quad-core con cuantro núcleos Cortex-A53 a 1,2 GHz y una GPU Mali 400MP2, con 2 GB de memoria LPDDR3 y 16 GB de memoria Flash para el sistema operativo. Se puede ampliar memoria insertando una terjeta SD de hasta 256 GB. ¿Quieres saber más sobre este proyecto? Lee este artículo o echa un vistazo a este vídeo.

Vale, entonces... ¿tiene sentido montar un aula de informática?

Puedes llamarlo aula de informática, espacio *Maker*... o como quieras. Se trata de trabajar con dispositivos libres que nos permitan jugar con sus tripas. La Raspberry está diseñada para montar varios sistemas operativos. Podemos incluso tener varias tarjetas SD con distintos SOs y cambiar de uno a otro simplemente cambiando de tarjeta.

Copyright 2026

¿Por qué montarla con Raspberry?

- Porque es **hardware libre**.
- Tal vez sea porque fue de la primera en llegar, pero cuenta con **la mayor comunidad entorno a su producto**, si la comparamos con otras como Orange Pi, Banana Pi, Odroid...
- El **puerto GPIO** nos permite conectar con el mundo físico.
- Cuenta con sistema operativo basado en Linux y optimizado para este dispositivo (**Raspbian**).
- El **precio** de cada ordenador, si aprovechas teclado y pantalla, puede ser de **56 €**, que incluirían:
- Raspberry 3 modelo B, carcasa, alimentación oficial y tarjeta clase 10 de 16 GB (*desde 50 € aprox*).
- Adaptador VGA - HDMI, si tus monitores no tienen entrada HDMI (6 € aprox).

Hagamos cuentas

Con los precios que hemos comentado hasta ahora, si quisiéramos renovar nuestro aula de informática que tiene 30 puestos deberíamos invertir... **¡2280 €!** No está mal para un espacio Maker ¿no? ;-)

Un poco de Historia

En 2011 se desarrolló la Raspberry Pi como ordenador de bajo coste para facilitar la enseñanza de la informática en los colegios, pero hasta 2012 no comenzó a fabricarse. La fundación recibe apoyos del laboratorio de informática de la Universidad de Cambridge y de Broadcom.

Ahora los jóvenes tienen contacto con la informática pero, aunque parezca paradójico, es muy raro que realmente se facilite el desarrollo de aplicaciones o de programas como en los primeros ordenadores personales, especialmente a los más pequeños. Los ordenadores que tenemos están orientados a tareas informáticas o de ocio, pero no vienen preparados con conexiones que posibiliten **“pequeños proyectos de hardware”** o con herramientas para aprender a programar o un lenguaje de programación. Este es el nicho que cubre (perfectamente) la Raspberry Pi.

Como vemos no hay ningún secreto en su precio, ya que fue diseñada con el fin de ser lo más barato posible y llegar al máximo número de usuarios.

Son varios los modelos que han aparecido hasta ahora y, sin entrar en mucho detalle de cada uno de ellos, mostramos a continuación una tabla resumen con las características de cada uno de ellos.

TABLA COMPARATIVA DE LAS DIFERENTES VERSIONES DE RASPBERRY *

Copyright 2026

	Raspberry Pi 1 Modelo A	Raspberry Pi 1 Modelo B	Raspberry Pi 1 Modelo B+	Raspberry Pi 2 Modelo B	Raspberry Pi 3 Modelo B
SoC (Chip)	Broadcom BCM2835 (CPU + GPU + DSP + SDRAM + puerto USB)[3]			Broadcom BCM2836 (CPU + GPU + DSP + SDRAM + Puerto USB)	Broadcom BCM2837 (CPU + GPU + DSP + SDRAM + Puerto USB
CPU:	ARM 1176JZF-S a 700 MHz (familia ARM11)[3]			900 MHz quad-core ARM Cortex A7	1.2GHz 64-bit quad-core ARMv8
Juego de instrucciones:	RISC de 32 bits				
GPU (Gráfica)	Broadcom VideoCore IV,,[61] OpenGL ES 2.0, MPEG-2 y VC-1 (con licencia),[59] 1080p30 H.264/MPEG-4 AVC[3]				
Memoria (SDRAM):	256 MiB (compartidos con la GPU)	512 MiB (compartidos con la GPU)[4]desde el 15 de octubre de 2012		1 GB (compartidos con la GPU)	
Puertos USB 2.0:	1	2 (vía hub USB integrado)[54]		4	

Copyright 2026

Entradas de vídeo:	Conector MIPI CSI que permite instalar un módulo de cámara desarrollado por la RPF			
Salidas de vídeo:	Conector RCA (PAL y NTSC), HDMI (rev1.3 y 1.4),[63] Interfaz DSI para panel LCD[64] [65]			
Salidas de audio:	Conector de 3.5 mm, HDMI			
Almacenamiento integrado:	SD / MMC / ranura para SDIO	MicroSD		
Conectividad de red:[5]	Ninguna	10/100 Ethernet (RJ-45) vía hub USB[54]	10/100 Ethernet (RJ-45) vía hub USB[66], Wifi 802.11n, Bluetooth 4.1	
Periféricos de bajo nivel:	8 x GPIO, SPI, I²C, UART[61]			17 x GPIO y un bus HAT ID
Consumo energético:	500 mA, (2.5 W)[5]	700 mA, (3.5 W)	600 mA, (3.0 W)	800 mA, (4.0 W)
Fuente de alimentación:	5 V vía Micro USB o GPIO header			
Dimensiones:	85.60mm × 53.98mm			

(* tabla modificada de [https://es.wikipedia.org/wiki/Raspberry_Pi
](https://es.wikipedia.org/wiki/Raspberry_Pi)**)****

Copyright 2026

1. Materiales necesarios: Hardware y software

Al igual que no necesitamos ser ingenieros de telecomunicaciones para usar un móvil, tampoco hace falta ser informáticos para sacarle mucho partido a nuestro aula. Además, si podemos "montar" la clase con nuestros alumnos, conseguiremos que se formen tanto en Hardware como en Software.

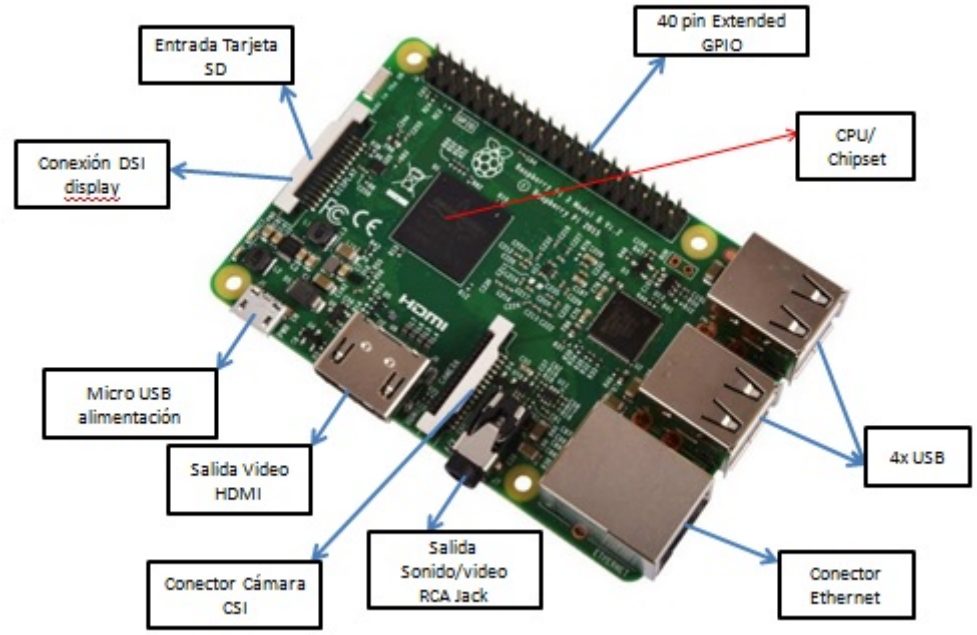
¿Por dónde podemos empezar?;

Pues está claro, conociendo físicamente nuestra Raspberry, sus partes y componentes fundamentales y porqué no, los multiples periféricos que le podemos añadir.

A.- HARDWARE:

Componentes principales de Raspberry Pi 3*:

Copyright 2026



(* Elaboración propia)

Periféricos necesarios para crear una aula de Informática:

Pues en realidad no demasiados aunque la elección entre unos u otros, puede facilitarnos mucho la tarea, nuestra Raspberry Pi (a partir de ahora solo nos referiremos al modelo 3) ya incluye elementos básicos como la toma de alimentación.

Copyright 2026



*

Para comenzar y si no es el caso que compremos una raspberry pi con el kit de inicio (algo altamente recomendable por precio y comodidad), necesitaremos:



berry Pi del polvo y de otros

*

*/ **Disipadores de calor:** Ayudan a nuestra Raspberry a no sobrecalentarse, se pegan muy fácilmente sobre los chipset de la placa.



*

***/ Tarjeta MicroSD:** La mayoría ya vienen con adaptador para ordenador de sobremesa o portatil, lo recomendable para este proyecto, sería una capacidad de al menos 16 GB pero sobre todo una calizamos en la misma tarjeta el





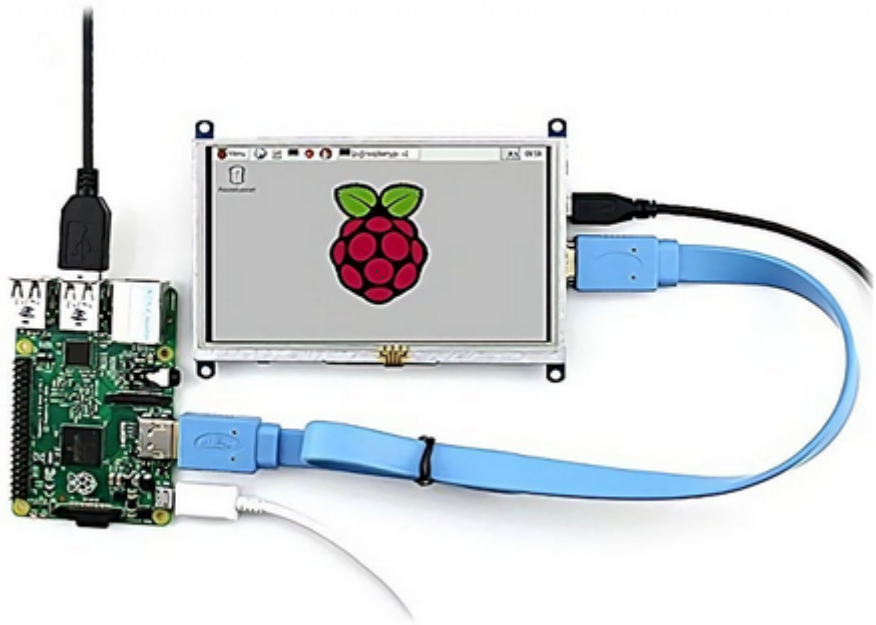
*

***/ Cable HDMI:** Para nuestro proyecto, no es necesario tener uno para cada Raspberry, aunque si en recomendable hacerse con unos cuantos.

(*Todas las imágenes extraídas de la Web de Amazon: www.amazon.es)

*/ **Teclado y ratón:** Aunque podrían servir cualquiera con entrada usb, si queremos que el aula sea funcional, estaría bien que se usase tecnología bluetooth. Si bien para una primera configuración es recomendable conectarlos mediante los USB, para evitar problemas de compatibilidad.

*/ **Touch Screen LCD:** Si la economía de nuestro centro nos lo permite, una buena solución que compagina los dispositivos de entrada y que resultará tremendamente cómodo y atractivo para nuestros alumnos es sin duda la "Touch screen" propia de Raspberry Pi. Compagina teclado y ratón sale rentable. Su precio va desde



*

Si exceptuamos esta última, los demás componentes si serán necesarios para montar nuestra aula de informática. A modo de sugerencia, Os imagináis un aula que mediante Wifi tenga los siguientes elementos para nuestros alumnos....

Touch screen mas caja (20€) + (Raspberry Pi3 B + Disipadores + Tarjeta SD + Alimentador + HDMI)(75 €) = - de 100€/alumno

(precio aproximado).

Qué más necesitamos para comenzar.... Pues un equipo de sobremesa que actuará a modo de servidor en nuestra aula. No es necesario que sea muy potente, de hecho cualquiera de los que tenemos en nuestros centros que ya no sirven para las Aulas convencionales puede servir.

*/ **Otros accesorios: La propia natulareza del proyecto Raspberry Pi, hace que el número de componentes que podemos añadir a nuestra placa no pare de crecer. Podemos**

encontrar alguna de los principales en los siguientes enlaces.

https://www.raspberryshop.es/accesorios-raspberry-pi.php

https://hipertextual.com/archivo/2014/05/accesorios-raspberry-pi/

http://www.raspipc.es/public/home/

Hasta cierto punto impresionante, ¿no?; Podemos ver miles y miles de proyectos que se han realizado con este mini ordenador, desde *consolas retro* hasta robots programables.

Recopilemos hasta ahora lo que necesitamos...

Hardware necesario

- Un equipo de sobremesa / portátil de edad para el servidor con un puerto Gigabit Ethernet.
- Un conmutador de red (requiere al menos un solo gigabit o 1000 puerto / 100 / 10mbit para el servidor).
- Un router (para una red independiente) o la conexión a la red de escuelas.
- Algunos cables Ethernet.
- Una Raspberry Pi y tarjeta SD con un tamaño de al menos 128 MB (recomendamos una tarjeta superior con al menos 4Gb de capacidad y velocidad de lectura-escritura +10.
- Tantas pantallas como raspberry queramos conectar. Podemos reaprovechar las de nuestro centro, usar touchscreen (quizá demasiado pequeñas para que los alumnos tengan una correcta visualización) o bien comprar pantallas nuevas.

Y ahora pasemos a Recopilar el software que vamos a necesitar, esta parte nos saldrá mucho más barata... de hecho es software gratuito y libre, por lo que solo tiene ventajas.

B.- SOFTWARE:

UBUNTU 16.04

Sistema Operativo Linux.

Para que nuestra aula de informática comience a tomar forma "digital" debemos previamente tomar una decisión, vamos a virtualizar el servidor esto es, sobre Windows XP o superior vamos a instalar VirtualBox y allí crear una máquina virtual con Ubuntu 16.04 o bien vamos a instalar desde cero en nuestro equipo servidor Ubuntu 16.04

Mi recomendación es la última de las opciones.

Así pues lo primero que necesitaremos será Ubuntu 16.04

Nosotros realizaremos el curso con ese Sistema operativo, eso sí, virtualizando el servicio mediante VirtualBox aunque todos los pasos son iguales que si lo realizásemos en una máquina propia.

La nueva versión de Ubuntu 16.04 LTS, es descargable desde: https://www.ubuntu.com/download

RASPBIAN:

Será el sistema operativo base que tendrán nuestros alumnos en sus Rapsberryv Pi.

PINET:

Pinet es un proyecto libre y de código abierto para ayudar a las escuelas establecer y gestionar un aula Raspberry Pi.
Se ha desarrollado junto a los maestros con la retroalimentación de más de 15 países de todo el mundo.
Sus características clave incluyen :

Que vamos a conseguir con ambos sistemas en funcionamiento:

Las cuentas de usuario de red : Las cuentas de los estudiantes de usuario se almacenan en el servidor central Pinet. Esto significa que pueden conectarse desde cualquier Raspberry Pi en el aula y no se pierde tiempo averiguando qué tarjeta SD pertenece a cada estudiante, o dicho de otra forma, las Raspberry Pi son intercambiables entre el alumnado.

El arranque de red: El sistema operativo Raspbian con el que Arranca la Raspberry Pi, también se almacena en el servidor central, es decir, cada vez que se enciende se restablece. Esto permite mantener un sistema operativo maestro perfecto, sin errores.

Carpetas compartidas y las copias de seguridad: Es muy sencillo configurar las carpetas compartidas para recursos con los estudiantes y también el uso de copias de seguridad automáticas.

Fácil de instalar : Pinet fue diseñado desde el principio para ser extremadamente fácil de configurar y mantener. De hecho son muchos los alumnos de menos de 12 años que han sido capaces de manejar a su antojo la red de equipos de este tipo de aulas.

Completamente gratuito: Pinet es totalmente gratuito y de código abierto. Esto significa que se puede utilizar para lo que quieras, donde quieras, sin coste y tener acceso a todo el código fuente para, si se desea, mejorar, cambiar o modifical cada uno de los aspectos que queramos, adaptándolo a nuestras necesidades.

Uso internacional: Este tipo de proyectos no son novedad en el mundo, muchos centros educativos de otros tantos países se han lanzado a "construir" este tipo de aulas, por lo que no estaremos solos en nuestra labor, de hecho sería beneficiosa la colaboración entre centros.

*/ TODAS LAS IMÁGENES UTILIZADAS EN EL PRESENTE MODULO HAN SIDO ELABORADAS POR EL AUTOR DEL CURSO.

Copyright 2026

2. Instalación de Ubuntu 16

INSTALACIÓN UBUNTU 16.04

Vamos a explicar mediante capturas y paso a paso el proceso de instalación de Ubuntu 16.04. en Virtual Box.

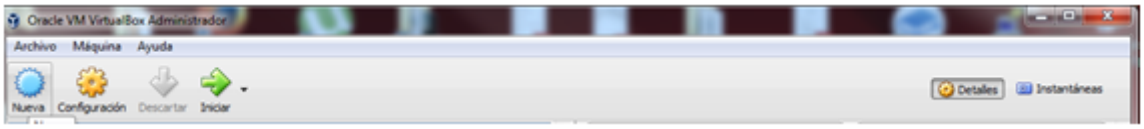
En el caso de que vosotros escojáis la opción de instalación directa en el equipo, deberéis continuar leyendo en el apartado de instalación del Sistema Operativo

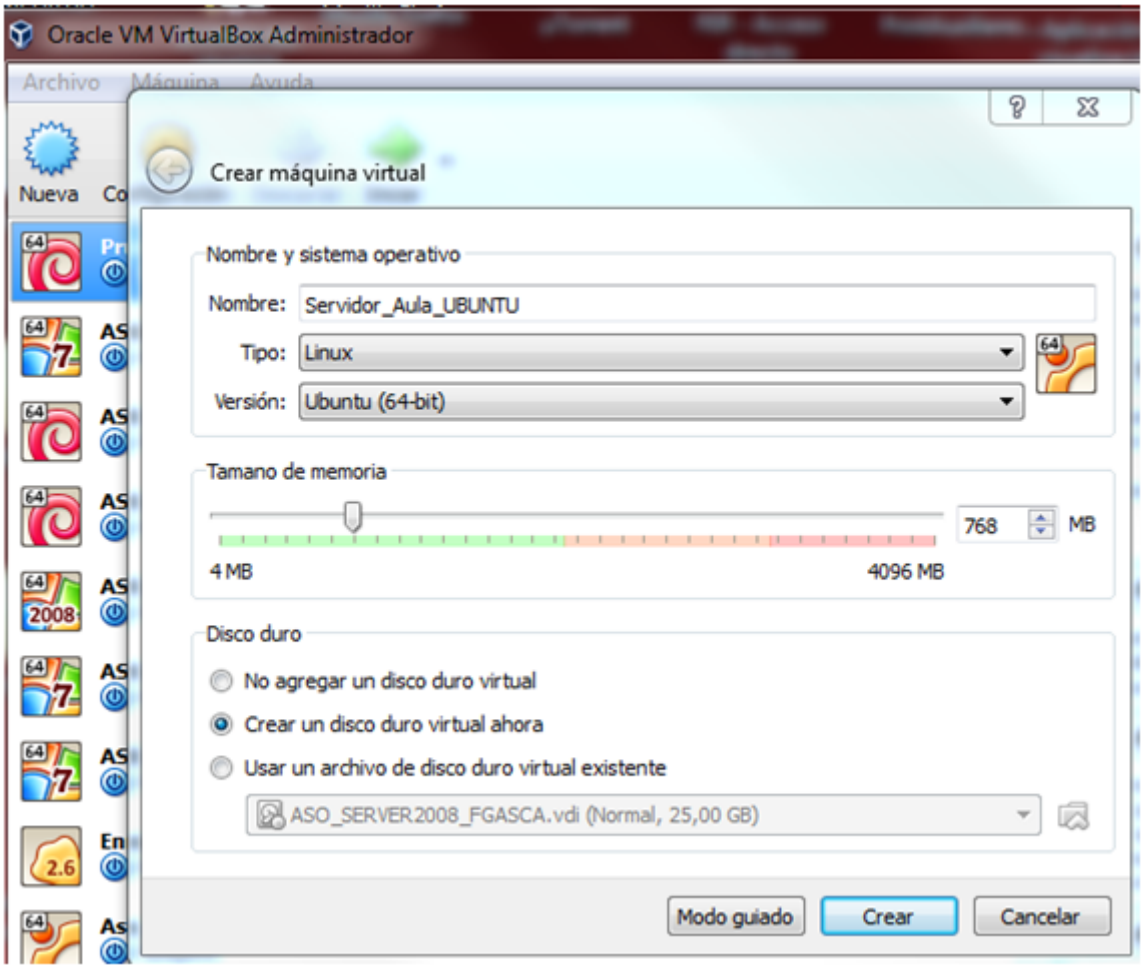
1.- CONFIGURACIÓN DE VIRTUAL BOX

Hemos elegido la versión de escritorio (16.04.02) desde la página oficial de ubuntu y descargaremos la imagen ".iso"

https://www.ubuntu.com/download/desktop/thank-you?country=ES&version=16.04.2&architecture=amd64

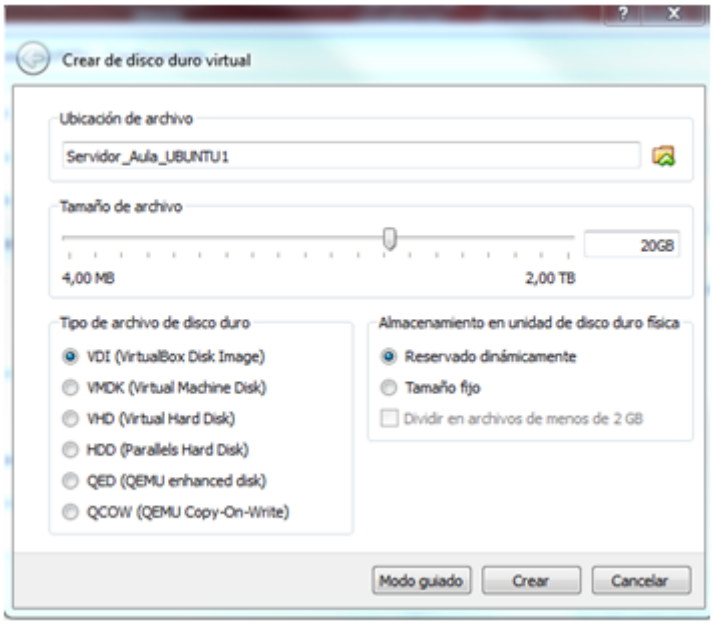
Abrimos virtualbox y pulsamos en nueva para agregar la nueva máquina virtual



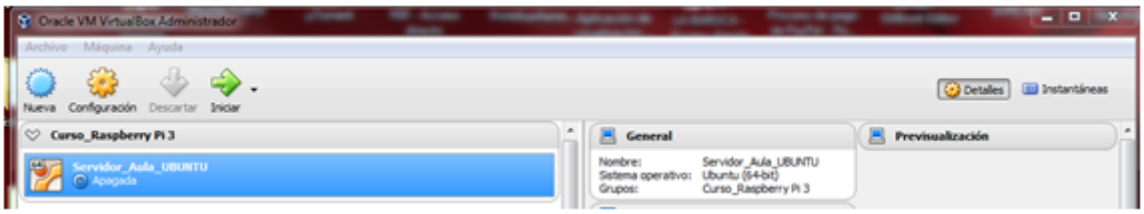


Como vemos le ponemos un nombre a nuestra elección y dejamos el resto de los elementos como nos aparece por defecto, si el ordenador sobre el que se implemente va a estar dedicado sólo a ser servidor del aula, podemos aumentar los parámetros como el tamaño de la memoria virtual o física, **Pulsamos en crear.**

Copyright 2026

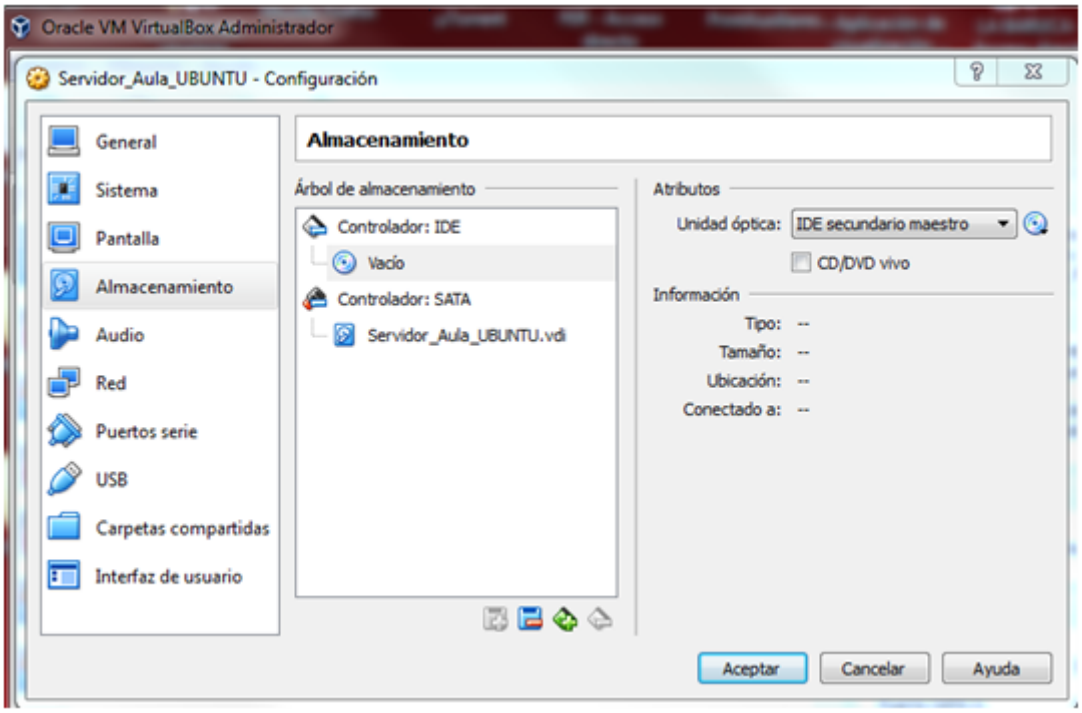


Configuramos ahora la memoria física (disco duro) que subimos al menos a 20.00 GB (Siempre podremos ampliarla posteriormente conforme más memoria se necesite). De nuevo **Pulsamos en crear**

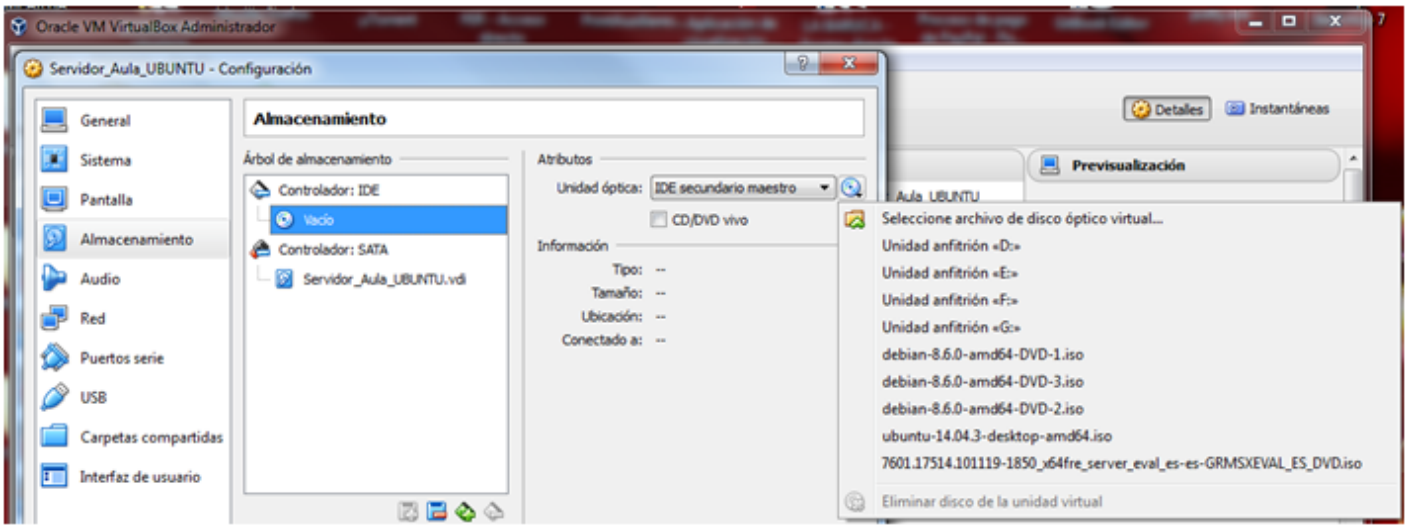


Ya la vemos creada y ahora pasaremos a configurarla, la seleccionamos y pulsamos sobre el botón **Configuración**. Para no alargarnos demasiado en la explicación, comentar que desde este lugar podremos configurar nuestra máquina en todo lo relativo a conexiones de red, inserción de imágenes con software, etc. Ahora vamos a ir un poco más al grano.

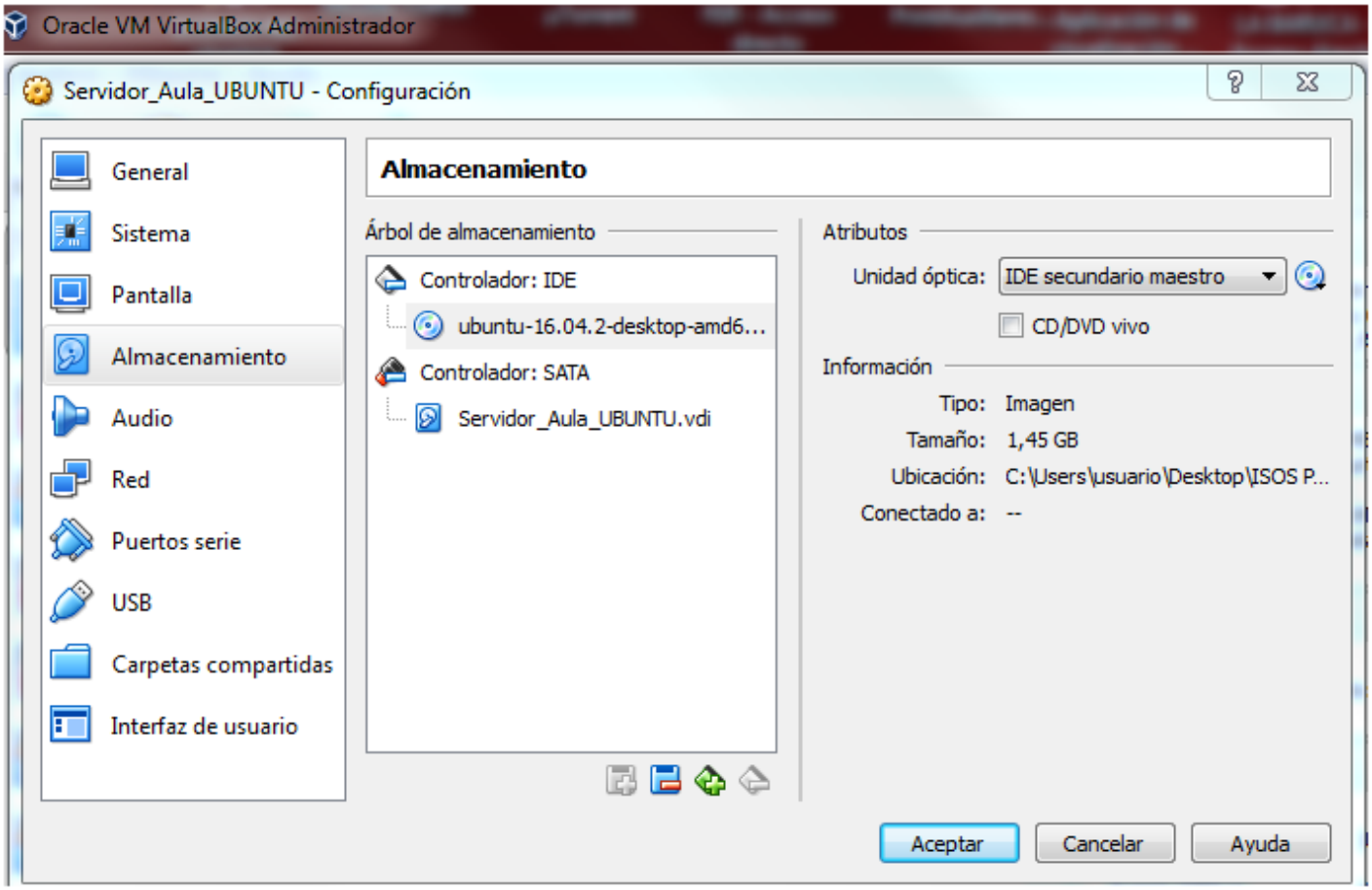
Vamos al menú **Almacenamiento** y Pulsamos sobre el disco DVD que aparece vacío para posteriormente pulsar en la otra imagen de la derecha que tiene ese mismo icono.



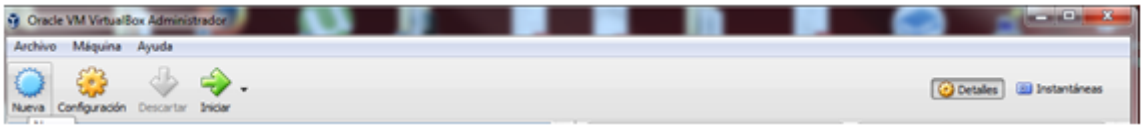
Al hacerlo se nos abrirá un menú en el que deberemos de buscar y seleccionar la Imagen .iso que nos acabamos de descargar.



Una vez seleccionada, veremos como nos aparece debajo del Controlador IDE

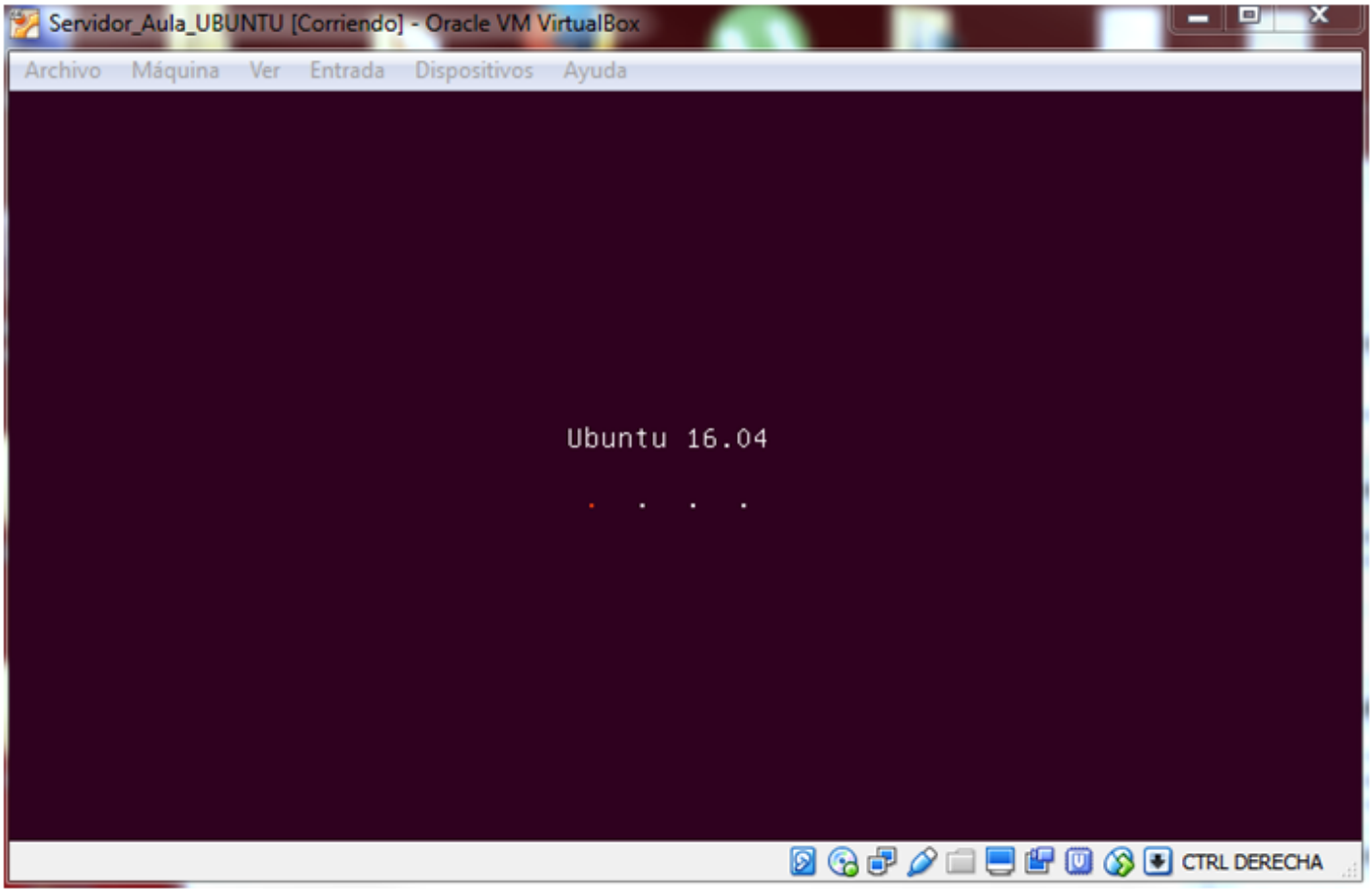


Aceptamos y ya no tenemos más que pulsar sobre el botón iniciar para que comience la instalación del sistema operativo.



Configurada ya nuestra VirtualBox, podemos pasar a Instalar el Sistema Operativo.

2.- INSTALACIÓN DEL SISTEMA OPERATIVO:

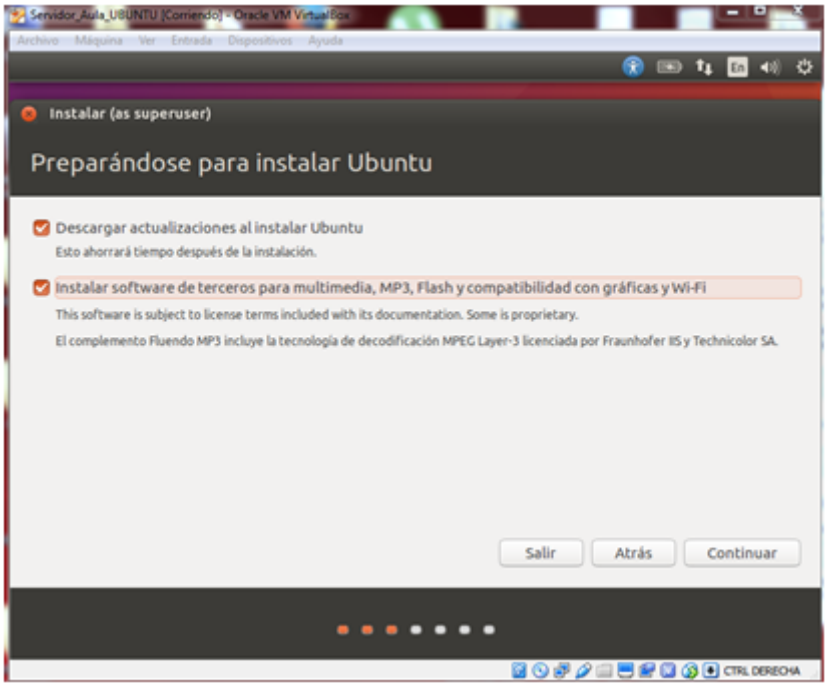


La instalación comenzará automáticamente y en principio no debería dar problemas.



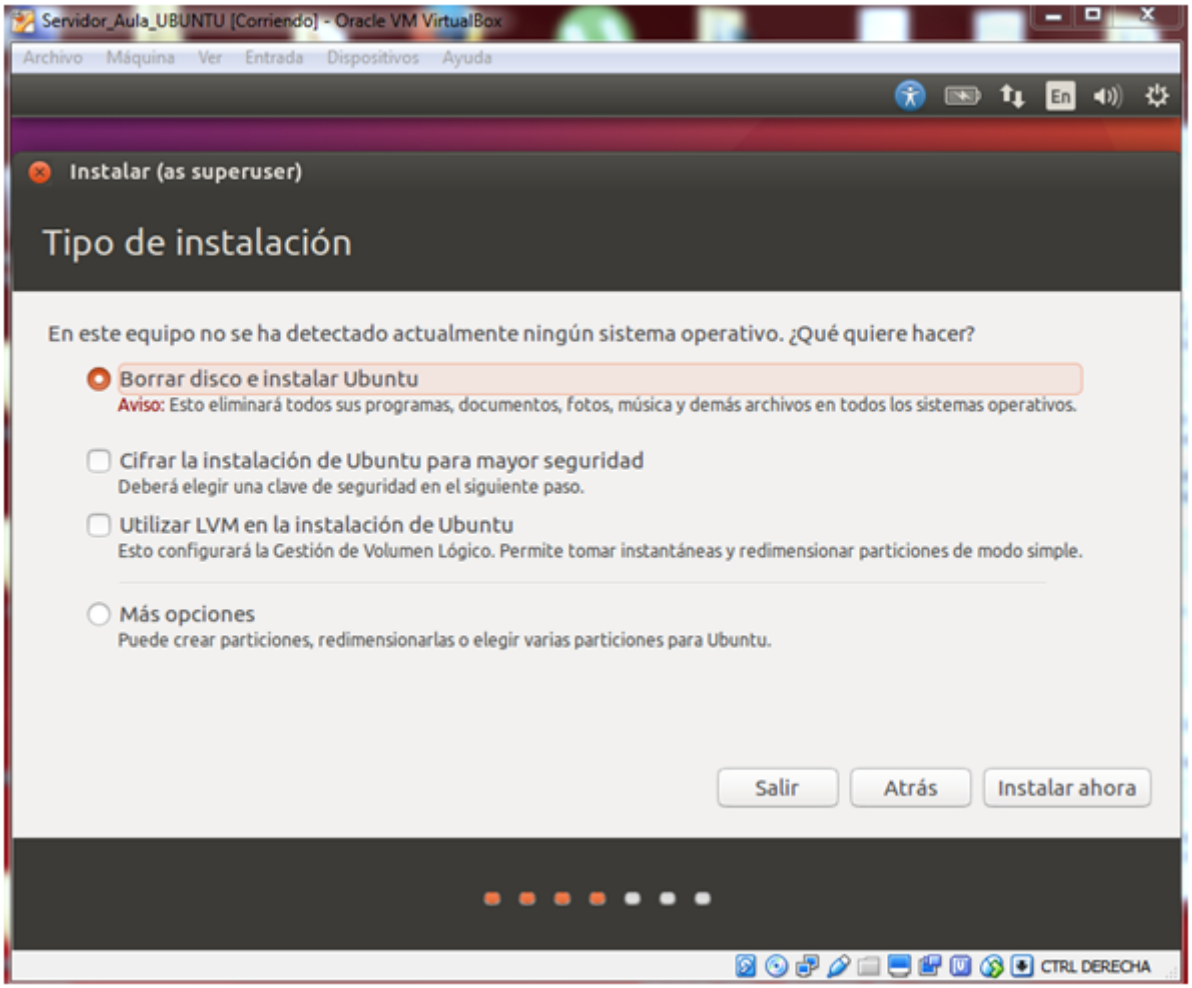
Seleccionamos el idioma y pulsaremos sobre **Instalar Ubuntu.**

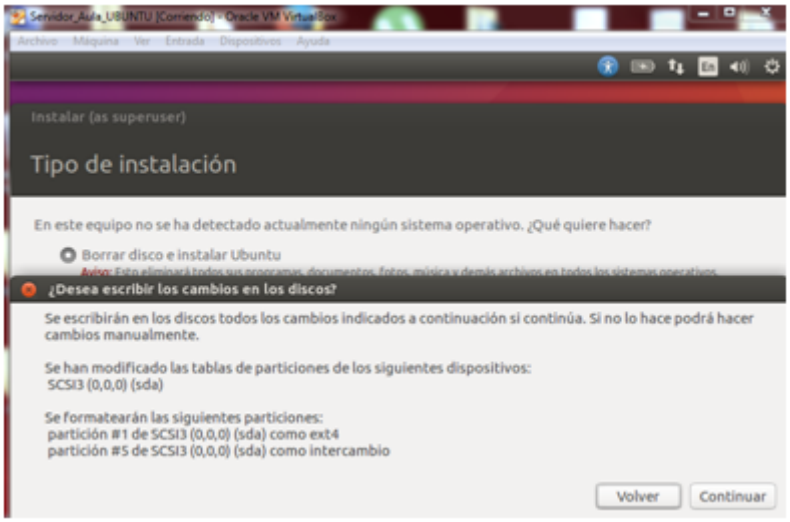
En la siguiente pantalla, y si la red es buena, recomendamos marcar ambas opciones, sino es así, es mejor dejarlo para realizar en tareas posteriores a la instalación.



Pulsamos continuar.

En la pantalla del tipo de instalación, dejamos la que nos aparece por defecto que formateará todo el disco duro e instalará el nuevo sistema operativo.





!!!!! ES OBVIO QUE ESTA OPCIÓN ELIMINARÁ CUALQUIER RESTO DE NEUSTRO ANTERIOR EQUIPO, INCLUÍDOS ARCHIVOS QUE PUDIERAMOS TENER ¡¡¡¡¡

Pulsamos sobre continuar y sobre instalar ahora.

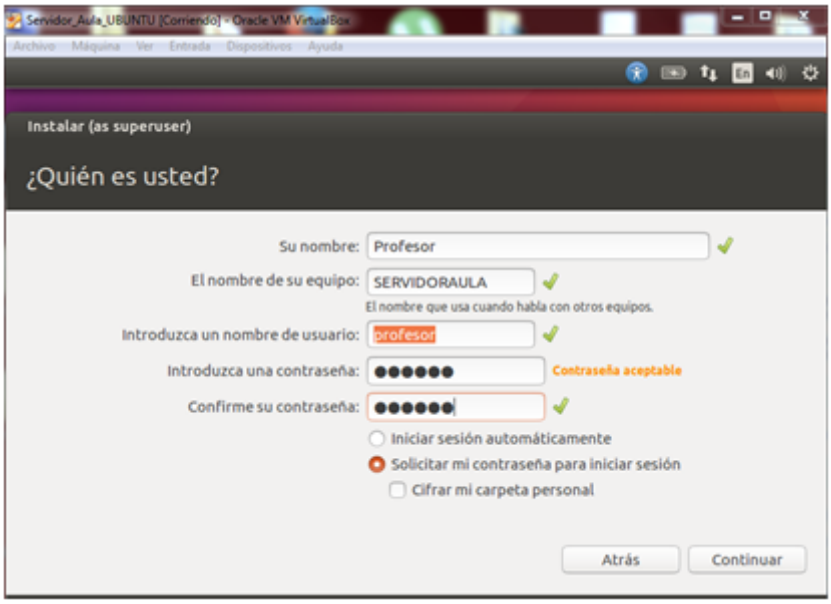
En las siguientes pantallas es cuestión de elegir las siguientes opciones:

Lugar: Madrid; Teclado: Español

Nombre de usuario y contraseña.

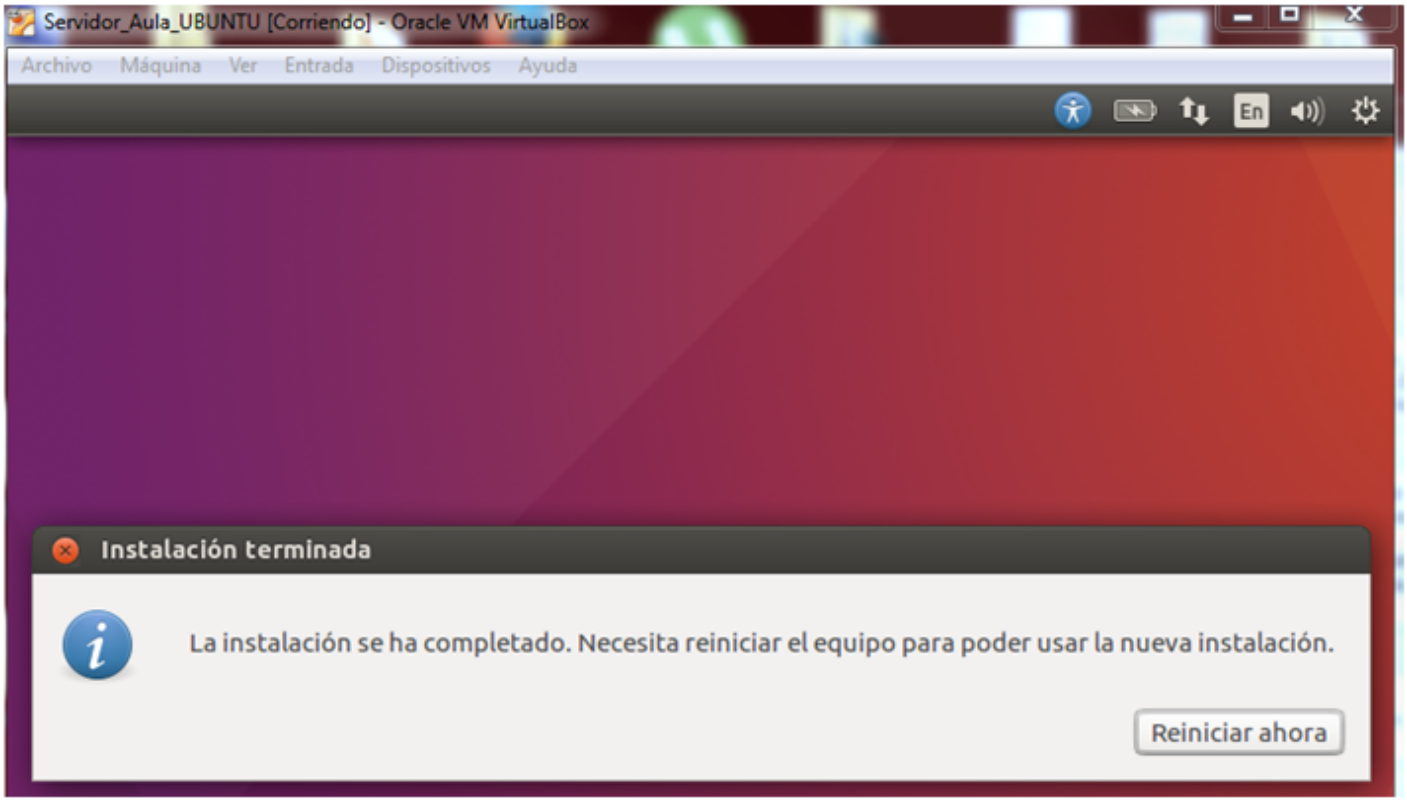
Nombre de equipo y del servidor.

Estas últimas opciones vendran determinadas por las necesidades de cada uno de los Servidores, número de personas que actuarán como admin, etc.



Una vez pulsemos en continuar, comenzará la instalación propiamente dicha por lo que tendremos que esperar (en función de las capacidades del equipo más o menos tiempo) mientras tanto siempre podemos observar alguna de las novedades que trae el nuevo sistema operativo.

Finalizado el proceso reiniciamos nuestro equipo que este listo para el siguiente Módulo



*/ TODAS LAS IMÁGENES UTILIZADAS EN EL PRESENTE MODULO HAN SIDO ELABORADAS POR EL AUTOR DEL CURSO.

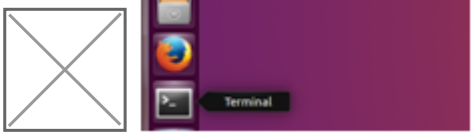
3. Instalación de PiNet

(NOTA DEL AUTOR): Recomiendo realizar todo el módulo sin entrar como root en nuestro servidor Ubuntu, ya que podríamos tener problemas con los permisos de las carpetas que crea PiNet al iniciar.

Una vez que tenemos un equipo con Ubuntu 16.04 o posterior podemos pasar a instalar PiNet, el programa que actuará como servidor de nuestra aula de Informática de Pi´s.

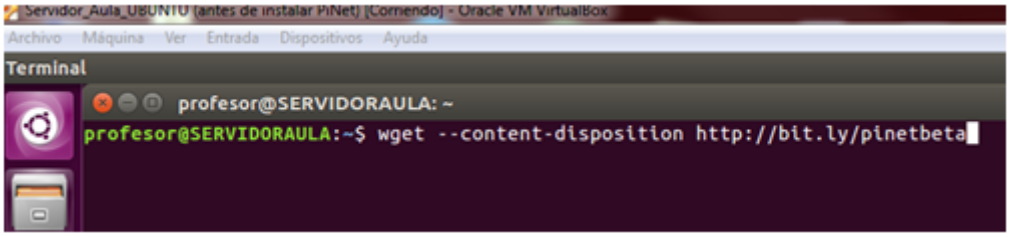
Su instalación resulta bastante sencilla;

*/ Comenzaremos por abrir nuestra terminal bien mediante el cuadro de buscar que aparece al pulsar en el botón de menú de Ubuntu o bien mediante el acceso directo a programas favoritos que aparecerá a la izquierda

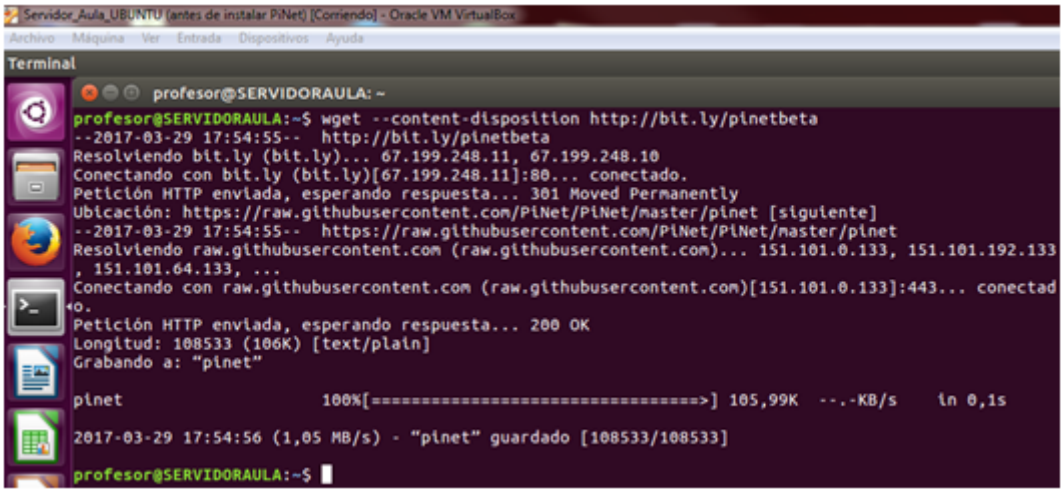


Una vez abierto escribiremos el siguiente comando (respetemos guiones y barras):

```
wget --content-disposition http://links.pinet.org.uk/jessie-stable-pinet
```

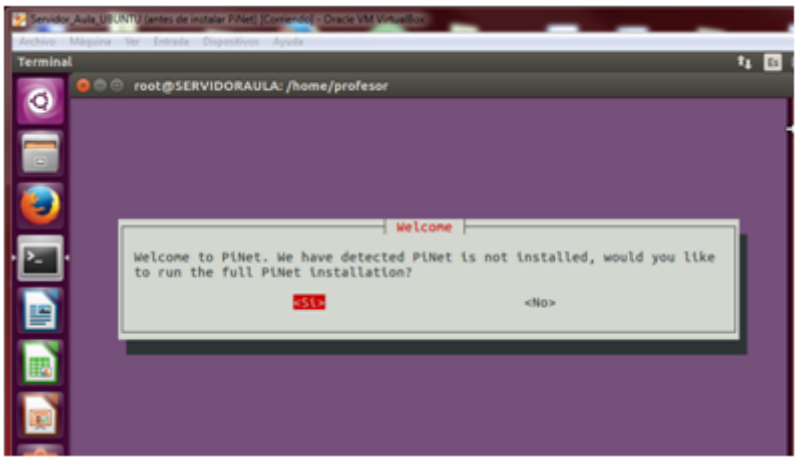


Nos descargará el instalador en nuestro equipo

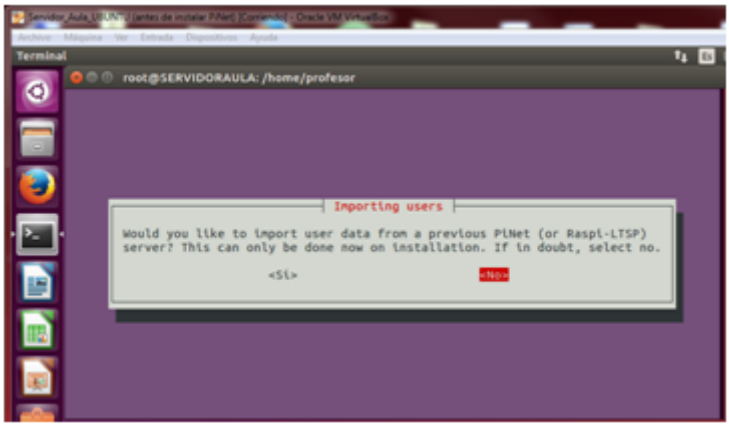


Deberemos ahora "arrancarlo", mediante el comando

```
sudo bash pinet
```

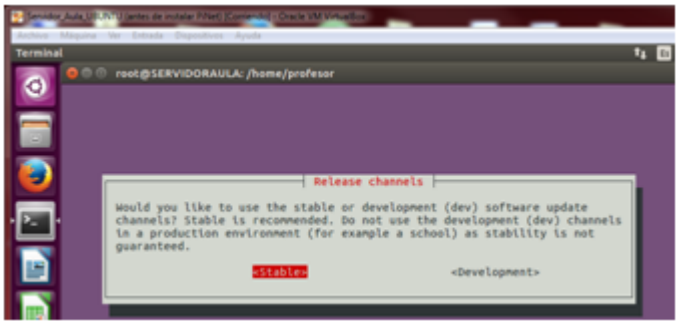


Marcamos SI

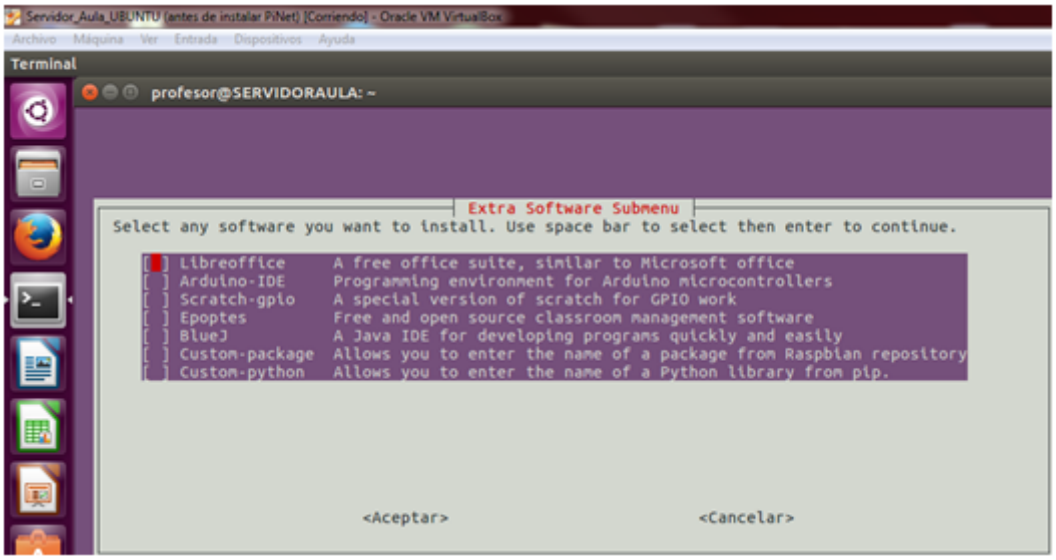


Marcamos No, ya que no tenemos usuarios creados con anterioridad, si queremos reinstalar PiNet podremos recuperar nuestro servidor (o incluso instalarlo en otro equipo pero con los mismos datos).

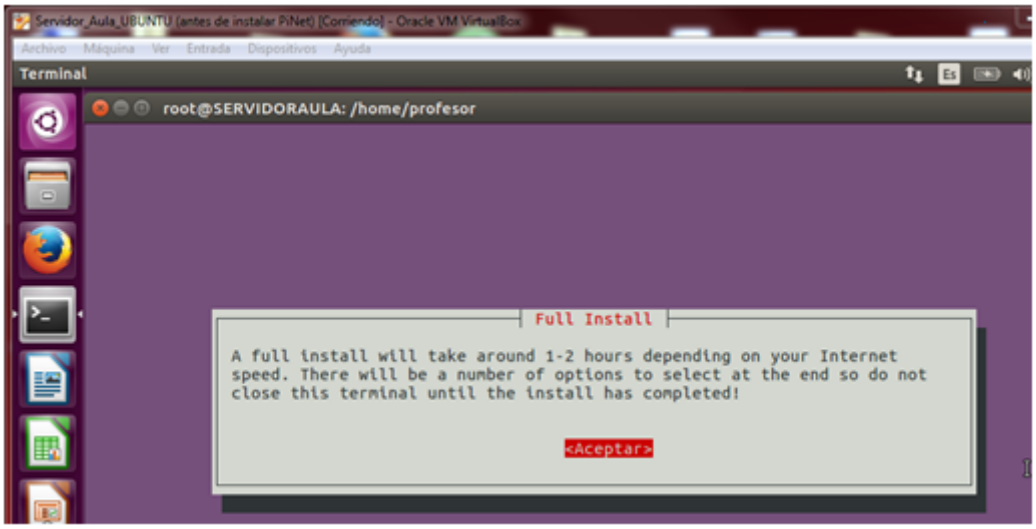
En la siguiente ventana nos preguntará que tipo de "Canales" queremos utilizar, marcaremos la opción "Stable"



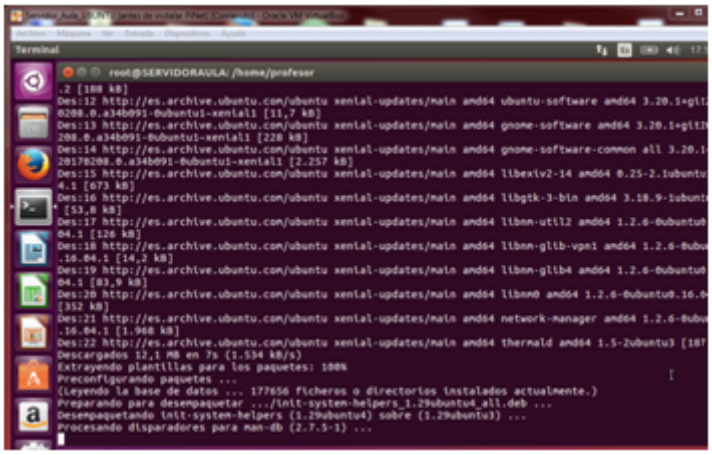
Nos pregunta después si queremos instalar algún software adicional, para este curso lo dejaremos en blanco para posteriormente mostrar como podemos instalar nuevos programas en nuestras raspberry Pi desde el servidor PiNet.



Lo dejamos en blanco y pasamos ahora a finalizar nuestra configuración de la instalación.

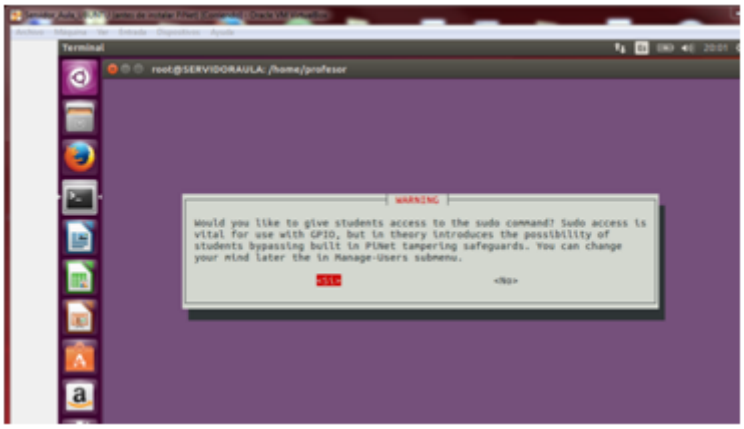


Cabe destacar que la duración dependerá tanto de la velocidad de internet como del número de paquetes que vayamos a instalar, si utilizamos VirtualBox aconsejamos subir la capacidad de Memoria RAM de nuestra máquina para acelerar el proceso, ya que puede llegar a durar hasta 3-4 horas.

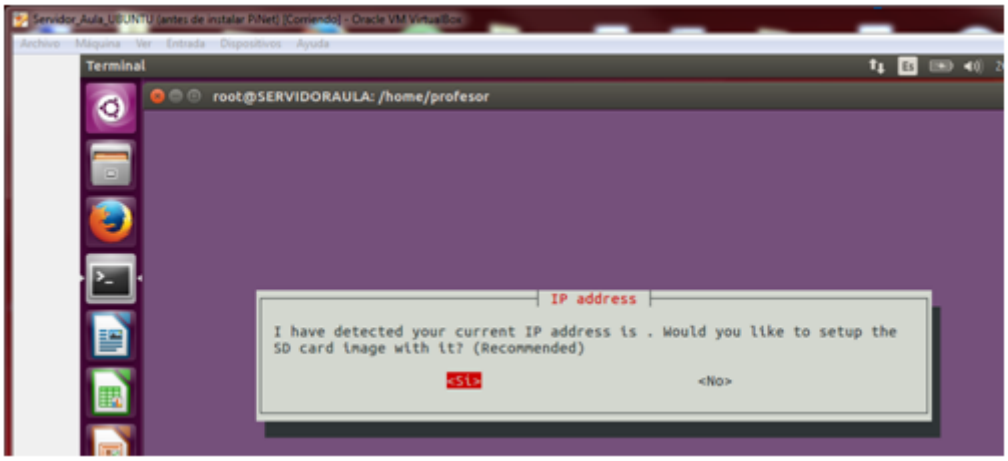

Imagen del proceso de instalación.

Una vez terminado, deberemos decidir si queremos que nuestros alumnos tengan acceso como root (sudo) o no. Ambas opciones tienen sus ventajas y desventajas y en cualquier caso podremos cambiarlas desde el menú de usuarios una vez que el sistema este instalado tanto en Ubuntu como en las Rasbperry Pi. En principio y salvo que queramos enseñar a manejar sistemas GNU/Linux, recomendamos impedir a los alumnos usar esta opción, por lo menos hasta que nuestro control del servidor sea suficientemente alto como para "deshacer" cambios indeseados.

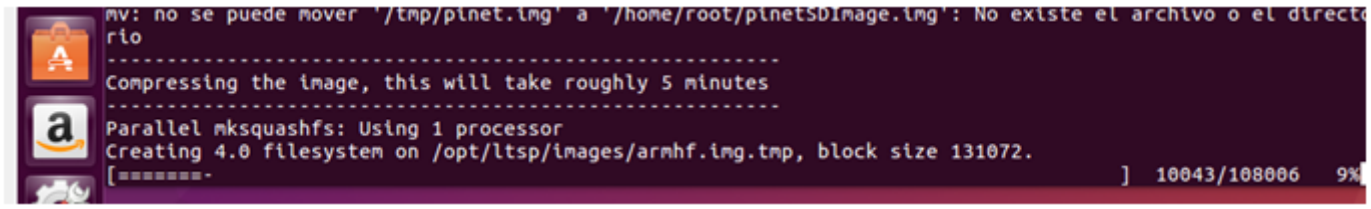
Copyright 2026



*/ Si sólo tenemos una red en nuestro servidor, dejaremos la siguiente opción como sale por defecto. En caso de que vayamos a crear una red única de Aula con el servidor actuando como tal y los equipos de los alumnos como “terminales tontos”, entonces deberemos elegir en el siguiente menú la red que vayan a usar nuestros alumnos. En nuestro caso, lo dejamos como sale (en Sí) La configuración podrá cambiarse más adelante.

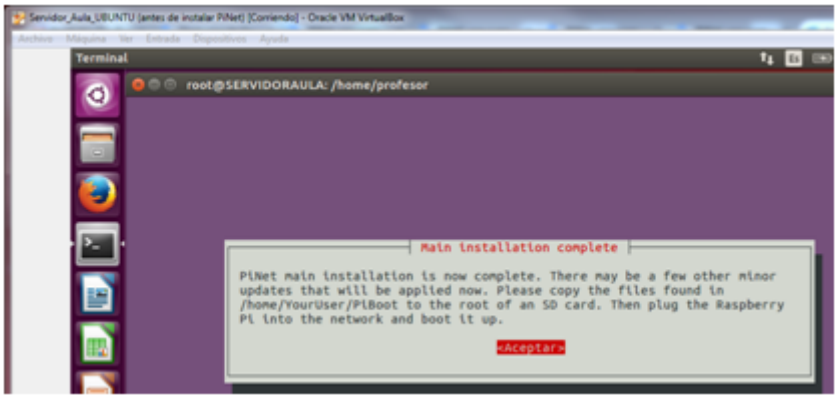


Configurada esta parte, pasaremos a ver un mensaje como el siguiente, nos acostumbraremos a ello, ya que aparecerá cada vez que cambiemos algo en PiNet.

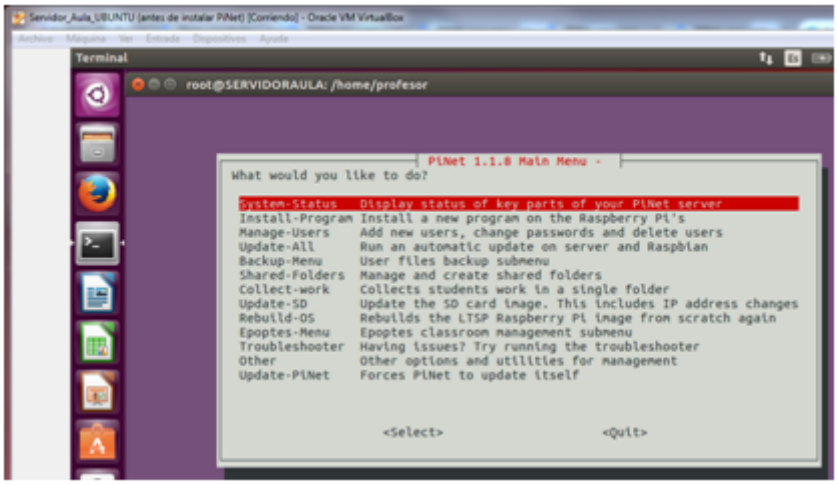


Aceptamos los dos siguientes mensajes sobre Información Adicional.

Y por último dejamos en blanco las preguntas que aparecen al final de la instalación.



Al aceptar nos arranca directamente el programa, aunque recomendamos quitarlo para pasar al módulo siguiente y configurar las SD para las Raspberry Pi.



*/ TODAS LAS IMÁGENES UTILIZADAS EN EL PRESENTE MÓDULO HAN SIDO ELABORADAS POR EL AUTOR DEL CURSO.

Posibles problemas durante la instalación, por Ricardo Fuster Andújar

> Varios compañeros, entre los cuales me encuentro, reportamos hace días en el foro un problema que surgía a la hora de instalar PiNet en la Raspberry Pi, consistente en que, una vez instalado PiNet en Ubuntu y pasados los ficheros de la carpera Piboot a la SD card, se producía el siguiente fallo:
>
> **/bin/sh: can't access tty; job control turned off (intransfer)**

En la sección de "Dudas" del foro hay una entrada que incluye una captura de pantalla. Tras indagar todo el día de ayer en internet he conseguido solucionar el problema, al menos el que me afectaba a mi, por que según he podido leer el comando /bin/sh: can't access tty; job control turned off (intransfer) no se refiere a un único problema, sino que alude a varios. Por lo tanto, es posible que la solución que aquí expongo no sirva para todos aquellos afectados por esta situación.

En mi caso, el problema estaba en la dirección IP establecida en el archivo cmdline.txt. Este archivo se genera durante la instalación de Pinet, queda alojado en la carpeta Piboot y es uno de los que debemos copiar en la SD card para iniciar Pinet en la raspberry. Una vez copiados todos los archivo de Piboot en la tarjeta SD, abrimos cmdline.txt con el editor de texto (abrimos el archivo que hemos copiado en la SD, no el que está en la carpeta Piboot de Ubuntu). Veremos lo siguiente:

dwc_otg.lpm_enable=0 console=serial0,115200 kgdboc=serial0,115200 console=tty1 init=/sbin/init-ltsp nbdroot=:/opt/ltsp/armhf root=/dev/nbd0 elevator=deadline rootwait

Después de nbdroot= debería aparecer nuestra dirección IP. Lo que debemos hacer es modificar el fichero manualmente introduciendo nuestra IP. Repito, la modificación debemos realizarla en el archivo cmdline.txt copiado en la SD card (de hecho, el alojado en Piboot es un archivo solo de lectura y no permite introducir cambios). Quedará así:

dwc_otg.lpm_enable=0 console=serial0,115200 kgdboc=serial0,115200 console=tty1 init=/sbin/init-ltsp nbdroot=192.168.1.137:/opt/ltsp/armhf root=/dev/nbd0 elevator=deadline rootwait

Por supuesto, la dirección IP que aparece aquí es la mía, de manera que debeís saber cual es la vuestra e introducirla. En ubuntu basta con abrir el terminal y teclear ifconfig. Después introduces la SD card en la raspberry y la enciendes. Si vuestro problema es el mismo debería resolverse. Espero que esta explicación os sirva de ayuda y podáis solucionar el problema.

4. Instalación del sistema en la Raspberry Pi

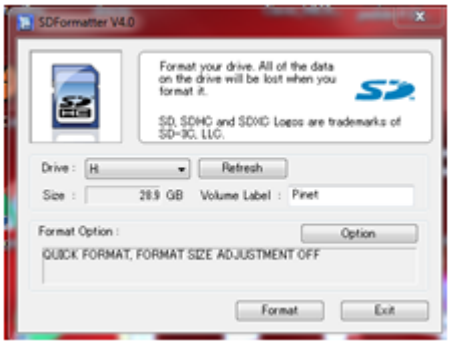
Si todo ha ido bien hasta ahora, y confiamos en que sí, ya tenemos preparado nuestro Servidor y podemos pasar a copiar los archivos necesarios en las tarjetas de memoria SD y proceder a unir nuestras Raspberry Pi.

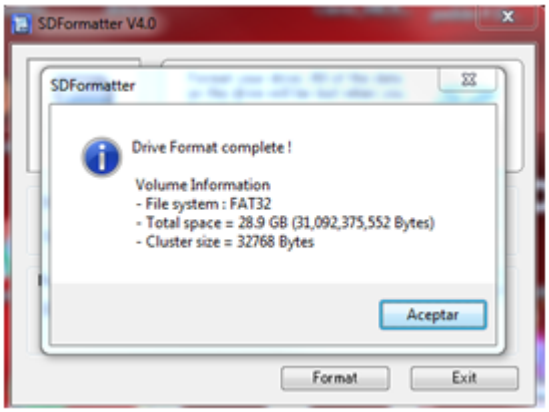
1.- FORMATEO DE LA TARJETA SD.

Deberemos formatear nuestra tarjeta SD (en formato FAT32) para evitar errores. Si usamos Windows recomendamos usar el Programa Gratuito **SDFormatter v4.0** que podemos descargar desde:

https://www.sdcard.org/downloads/formatter_4/index.html

Su uso es extremadamente sencillo, se inserta la tarjeta y nos aseguramos que la etiqueta del volumen (la letra) es la correcta en Drive,le ponemos nombre a la unidad (Volume Label) y pulsamos en **Format** y listo.





Si lo hacemos desde el propio Ubuntu u otro sistema Linux, podemos usar GParted que instalaremos mediante el comando:

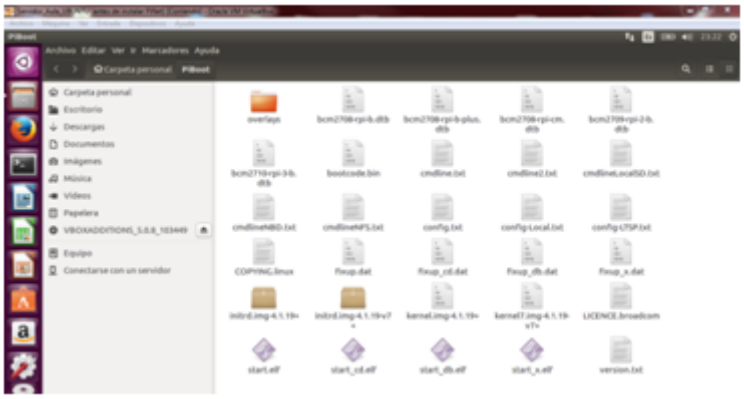
sudo apt-get install gparted

Para el resto de los pasos mediante este sistema, os enlazamos el siguiente tutorial online:

https://geekland.eu/formatear-memoria-usb-en-linux/

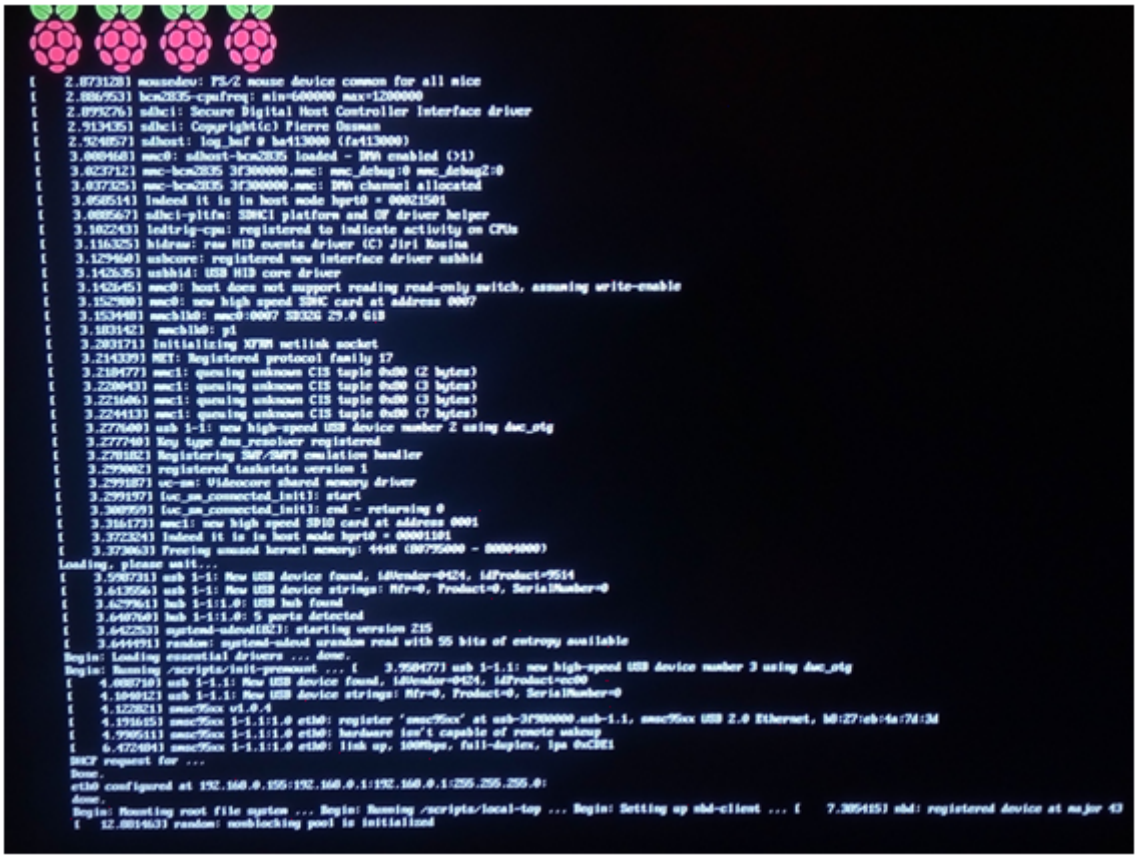
2.- PASAMOS LOS FICHEROS NECESARIOS A NUESTRA SD:

Para ello vamos en nuestro servidor Ubuntu a la carpeta piBoot (Carpeta personal PiBoot), seleccionamos todos los archivos y los pegamos en la tarjeta SD.



Ahora tan sólo deberemos insertar la SD en nuestra Raspberry Pi y encenderla (mediante el cable de alimentación y todos los cables de periféricos y red conectados).

Si todo ha ido bien, conezará la instalación (es posible que de algún problema de "error de entorno de escritorio" que no deberemos tener muy en cuenta, ya que sólo pasará con usuarios creados en el servidor, no pasará cuando tengamos alumnos creados).



Al cabo de unos minutos (el proceso puede tardar 30 minutos) podremos ver ya nuestra pantalla de Inicio de Raspbian.



Podemos aprovechar para familiarizarnos con el entorno de escritorio de este sistema operativo e incluso usar alguno de los multiples programas que vienen por defecto instalados y que veremos más adelante.

El curso continúa con la Creación de usuarios (alumnos).

***/ TODAS LAS IMÁGENES UTILIZADAS EN EL PRESENTE MÓDULO HAN SIDO ELABORADAS POR EL AUTOR DEL CURSO.**

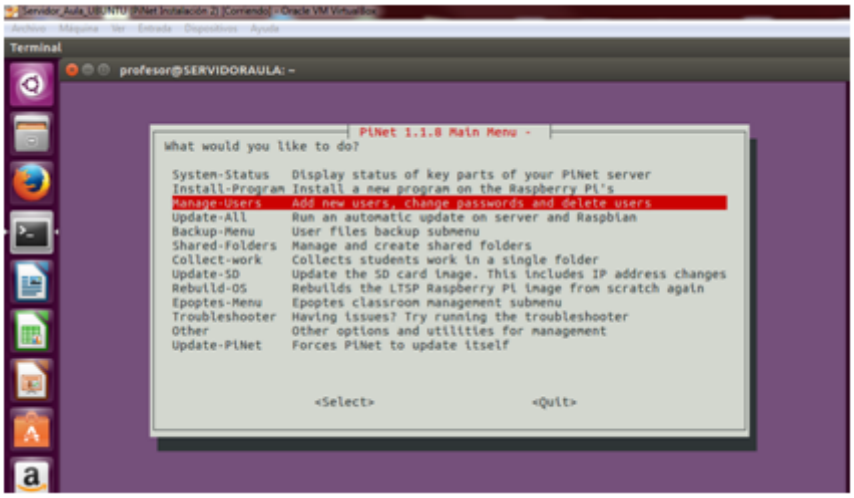
Copyright 2026

5. Creación de usuarios. Creación de Carpetas Compartidas

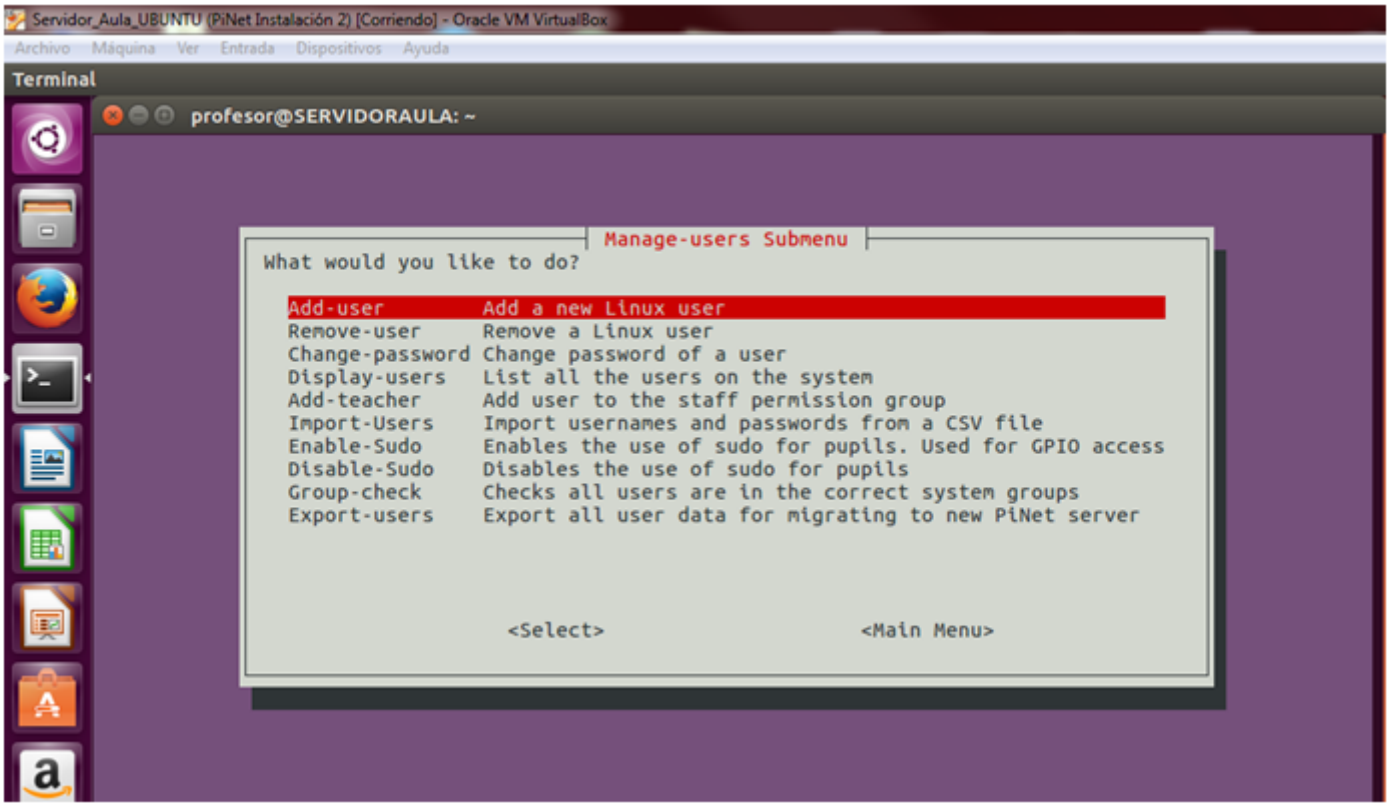
1.- CREACIÓN DE USUARIOS:

Cuando tengamos que crear nuevos usuarios lo realizaremos también desde el propio PiNet, esto nos permitirá que cada profesor y cada alumnado, tenga su configuración determinada. En principio los alumnos deberían poder acceder a sus terminales sin problemas de acceso y no tener un equipo para cada uno de ellos, sino que pueden utilizar el que deseen que guardará sus configuraciones.

Arrancamos PiNEt y vamos a Manage-Users



Nos aparecerá el siguiente menú:



Que por orden nos permitirá:

Por orden:

***Añadir nuevos usuarios**: Principalmente serán alumnos.

* **Eliminar usuarios:** Pues eso mismo

***Cambio de las contraseñas:** Muy útil en caso de pérdida por parte de los alumnos.

***Listar usuarios**: para ver todos los creados hasta ahora.

* **Añadir Profesor:** Aquí se nos permite añadir al grupo con más privilegios en el sistema, excepto el “root” del servidor a los usuarios creados mediante la primera opción.

* **Importar usuarios**: sencillo si tenemos algunos alumnos ya en un fichero .csv.

* **Enable –Sudo // Disable – sudo:** Permitir o denegar opciones de administrador de Raspbian a los alumnos. Puede ser beneficioso si queremos que los alumnos accedan al control de GPIO de sus Raspberry para realizar determinadas tareas o bien se instales sus propios programas.

* **Chequeo de grupo**: ver si están bien configurados.

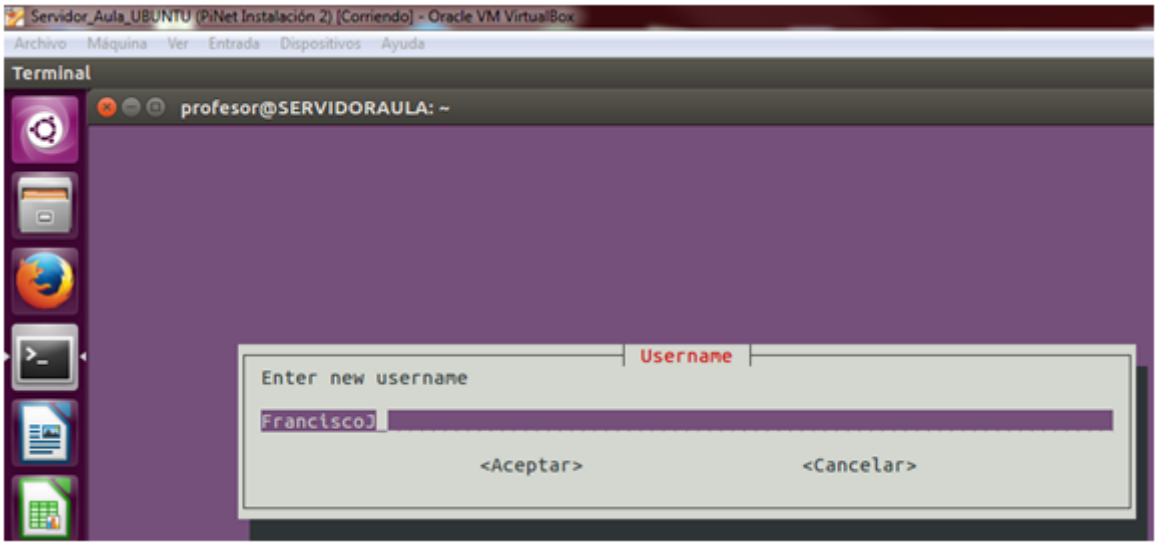
* **Exportar usuarios:** Configura un .csv que permite guardar los usuarios para una migración hacia un nuevo PiNet.

2.- PRACTICAMOS:

Practicaremos creando un nuevo alumno en nuestro caso FranciscoJ.

Vamos al menú explicado anteriormente y pulsamos sobre Añadir nuevos usuarios.

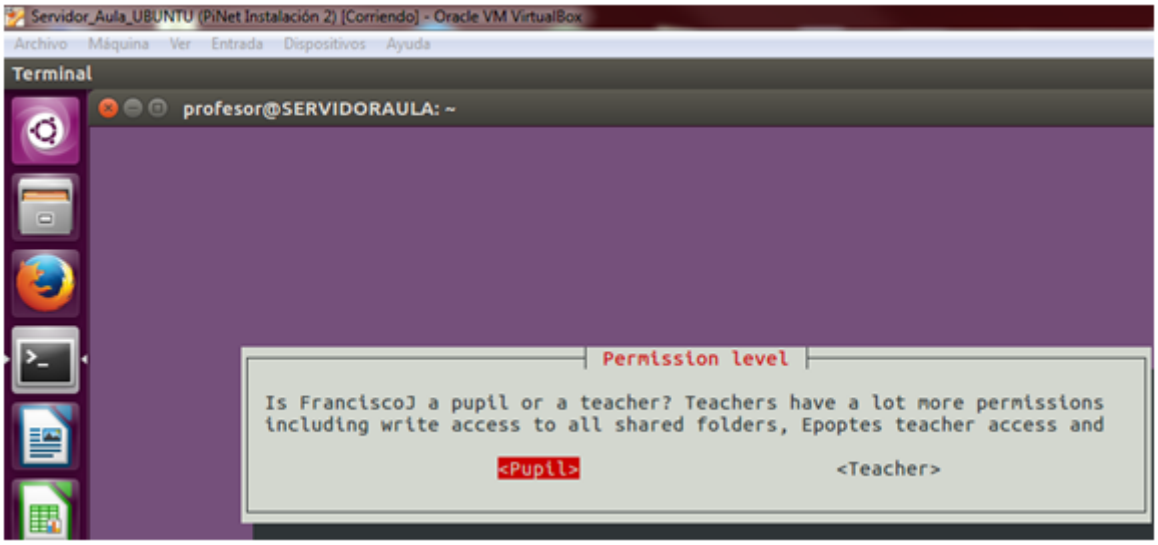
Introducimos su nombre



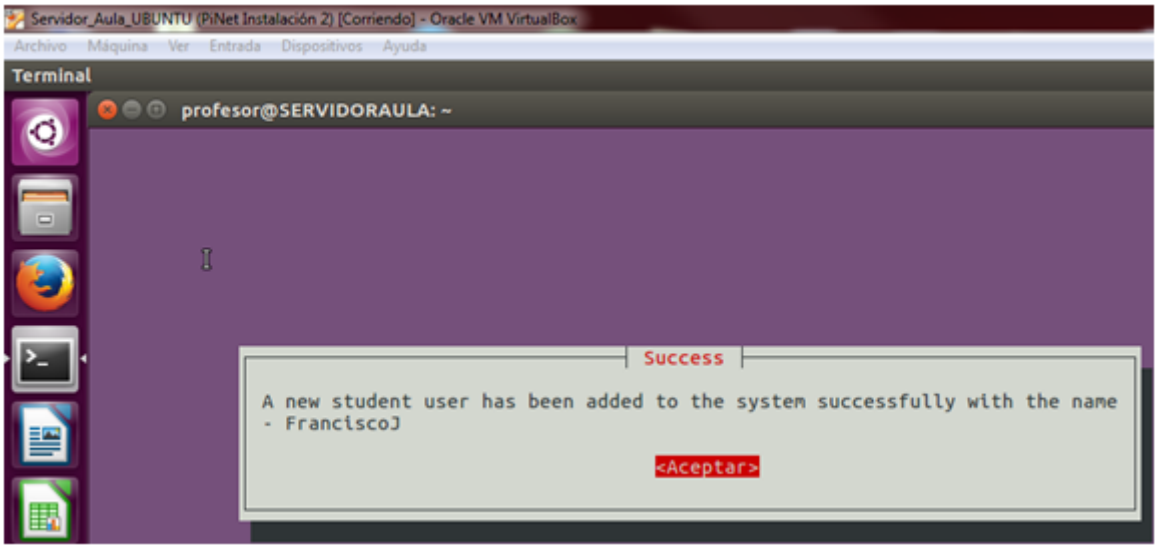
A la par ponemos una contraseña



Lo señalamos ahora como pupilo (alumno) o profesor



Ya lo tenemos creado

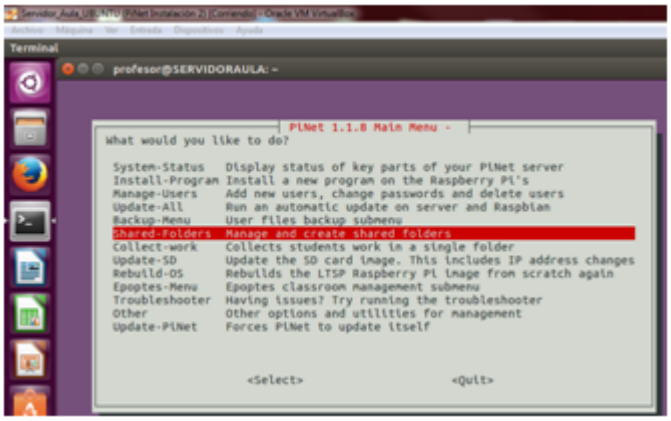


Nos preguntará si queremos añadir otro.

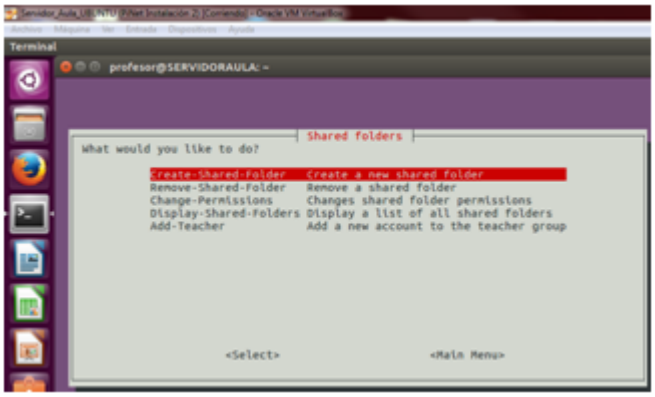
!!!!!Sencillo verdad¡¡¡¡¡¡ como más aprenderemos será llevándolo a la práctica así que debemos realizar la tarea encomendada para este módulo.

2.- CREACIÓN DE CARPETAS COMPARTIDAS:

Con PiNet arrancado, iremos a "Shared Folders"



Leeremos el Menú:



Vemos como podemos:

1.- Crear nuevas carpetas

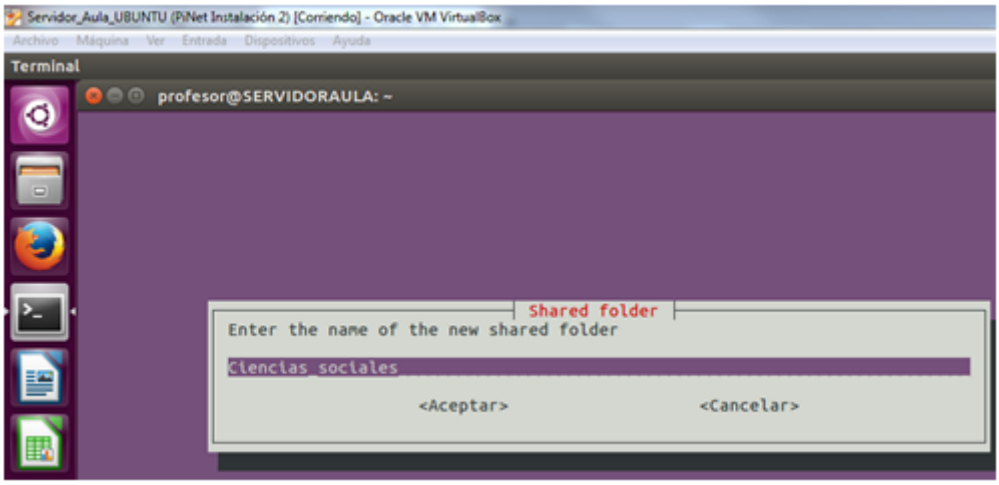
2.- Borrar carpetas compartidas con anterioridad

3.- Cambiar los permisos; Dar lectura-escritura a los alumnos, solo lectura...

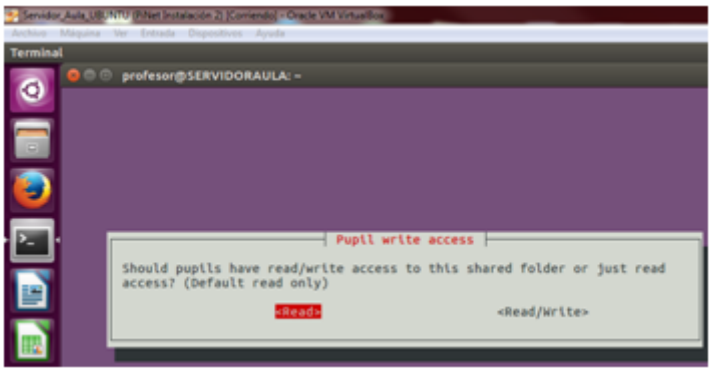
4.- Ver las carpetas compartidas en la actualidad.

5.- Añadir un profesor al grupo de profesores (necesario para compartir).

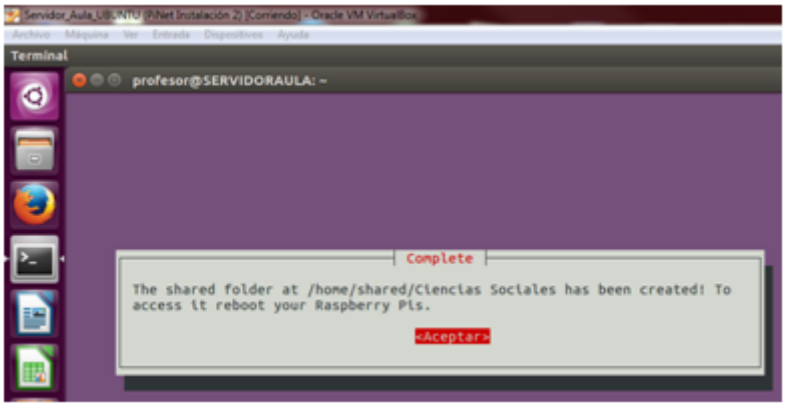
Elegimos la primera de las opciones y nos avisará que estas carpetas estarán accesibles desde todas las Raspberrys Pi; Marcamos que sí e introducimos el nombre de la carpeta compartida (sin espacios)



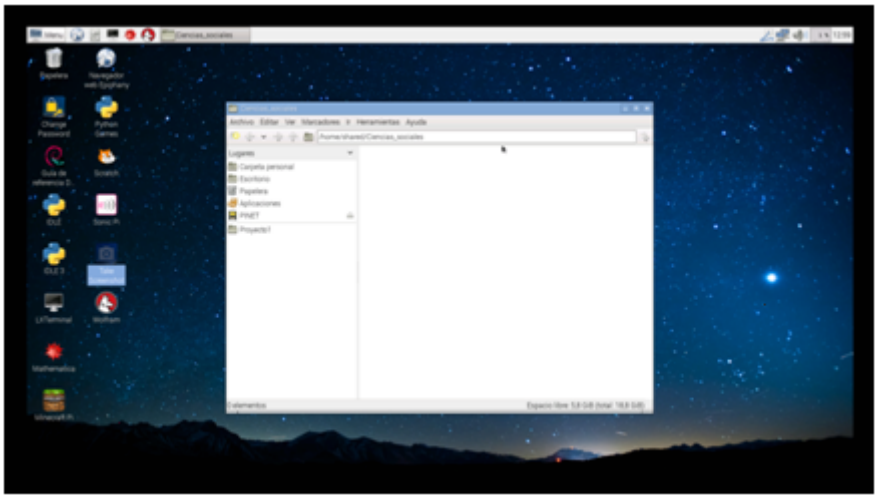
Otorgamos permisos a los alumnos, recomiendo leer y escribir, para que también puedan compartir recursos con el profesorado, pero también puede ser buena opción la primera para compartir apuntes, tareas, etc y que no puedan ser eliminados. En definitiva, deberemos adaptar los permisos a nuestros usos particulares de los recursos compartidos.



Nos mostrará la ruta al recurso y nos avisa que deberemos reiniciar los terminales para que todos los usuarios las vean.



Muestro como verá el alumno el recurso compartido que le aprecera directamente en sus carpetas



Mientras que el profesor lo verá así.



Os animo a que practiqueis esta parte del curso, comprobando si podemos mediante grupos de usuarios, tener acceso a unos recursos compartidos y no a otros, consiguiendo con ello evitar los problemas de "me han borrado mi trabajo" que todos hemos sufrido alguna vez.

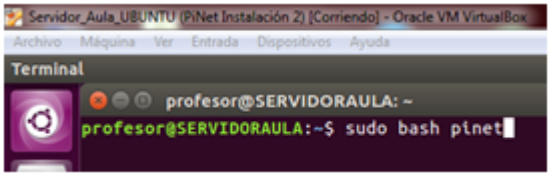
***/ TODAS LAS IMÁGENES UTILIZADAS EN EL PRESENTE MÓDULO HAN SIDO ELABORADAS POR EL AUTOR DEL CURSO.**

Copyright 2026

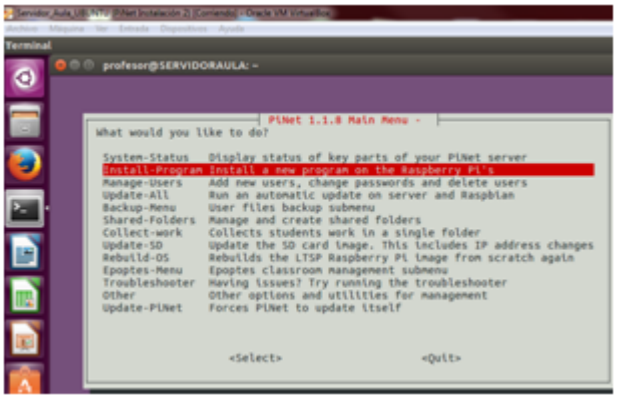
6. Instalación de un nuevo programa: “ Epoptes”

Nuestro Raspbian ya viene con bastantes programas listos para trabajar (posteriormente en otro capítulo los trataremos más en detalle), pero no así nuestro servidor, al que podremos instalar directamente desde Pinet algunos programas tan interesantes como el que os presentamos a continuación: EPOPTES.

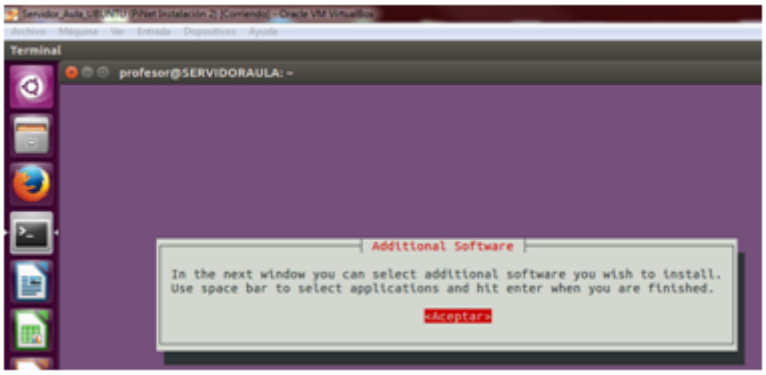
Arrancamos pinet:7



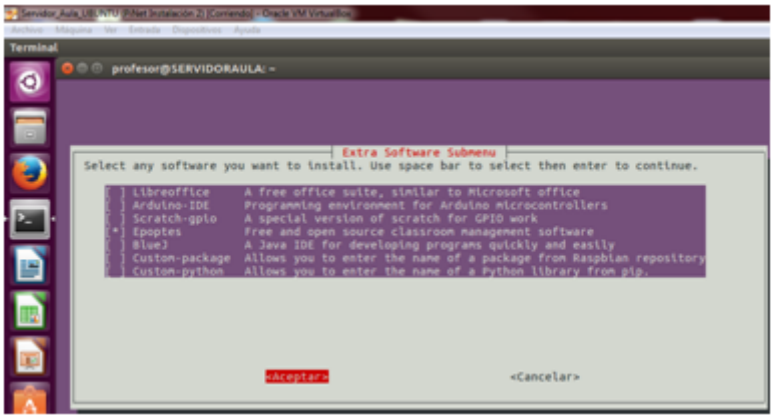
Vamos a _Install - Program _de nuestro Menú

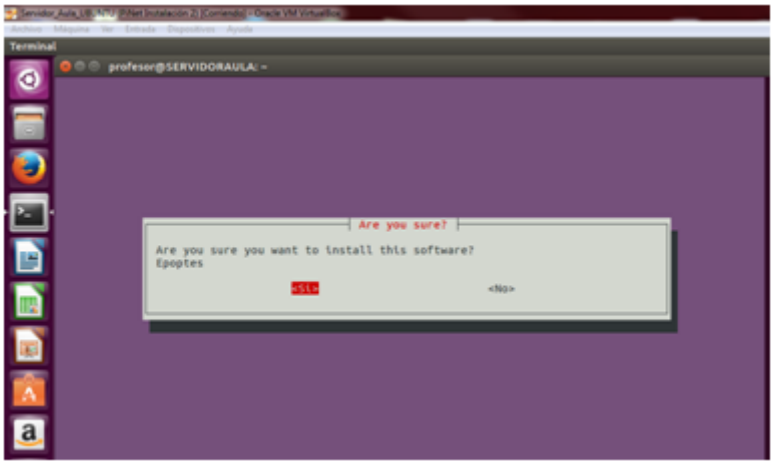


Aceptamos el mensaje de confirmación.

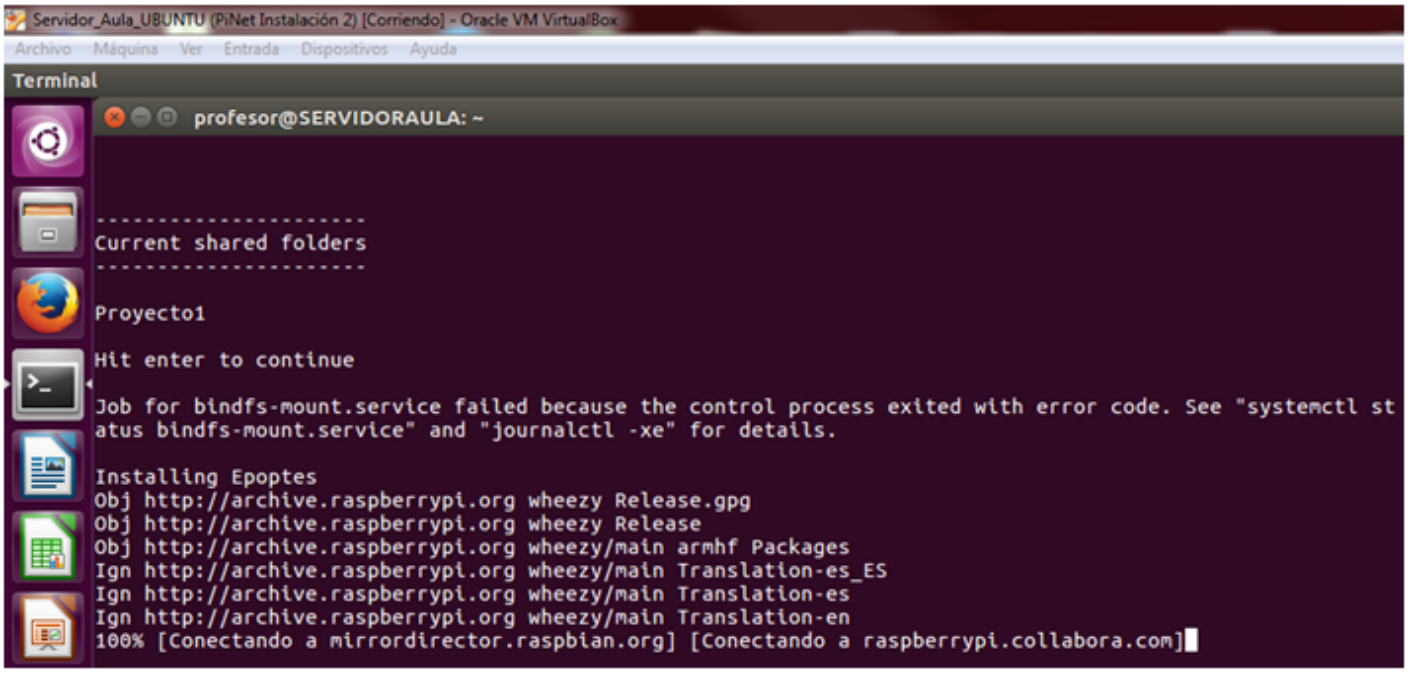


Marcamos el programa que nos interesa instalar (vemos como la pantalla es igual que la que obteníamos en la primera instalación de PiNet) y aceptamos.



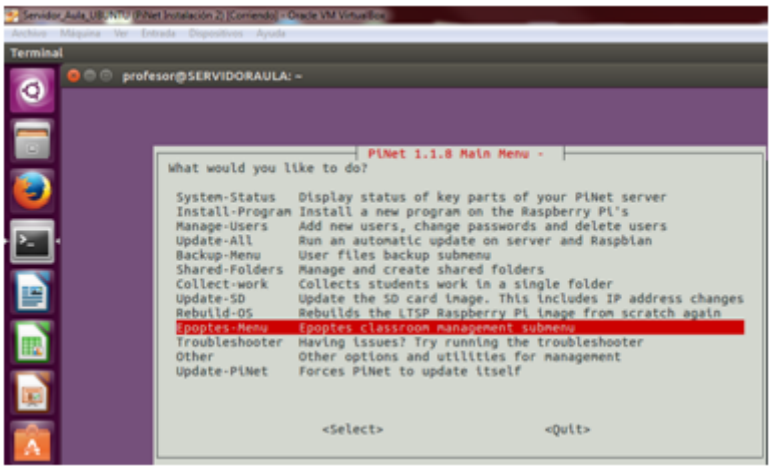


Aceptamos el mensaje de aviso y comenzará a instalar.

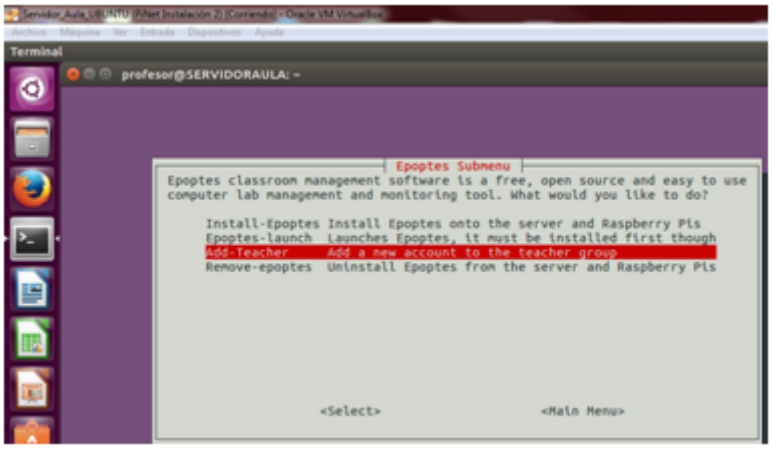


Vemos conforme avanza el proceso que crea una serie de carpetas compartidas que al reiniciar las Raspberry, creará y permitirán que sin modificar la Imagen de nuestra SD primaria, podamos utilizar el programa con todos nuestros terminales.

Una vez finalizado el proceso, si queremos usarlo tendremos que ir a *Epoptes - Menu*



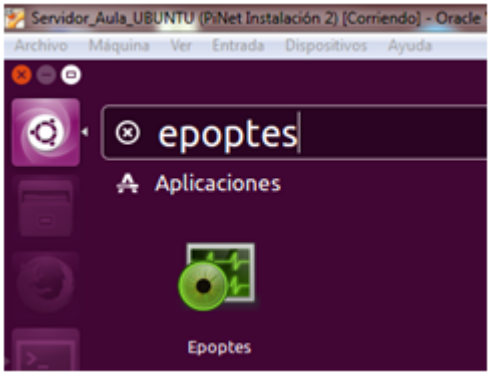
Deberemos añadir un profesor al grupo "profesores".



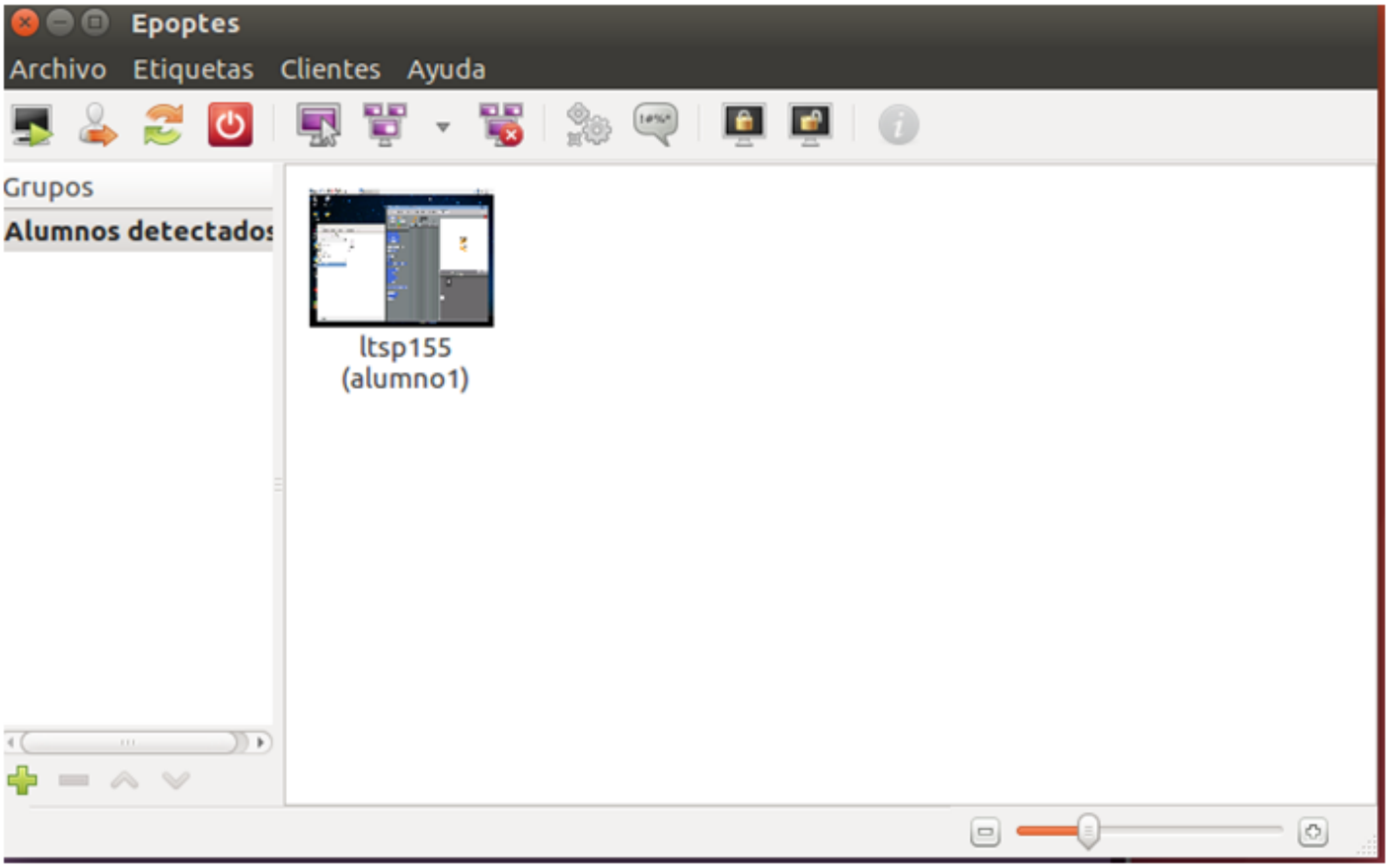
Lo cual nos permitirá que este usuario tenga acceso total a las opciones del servidor.



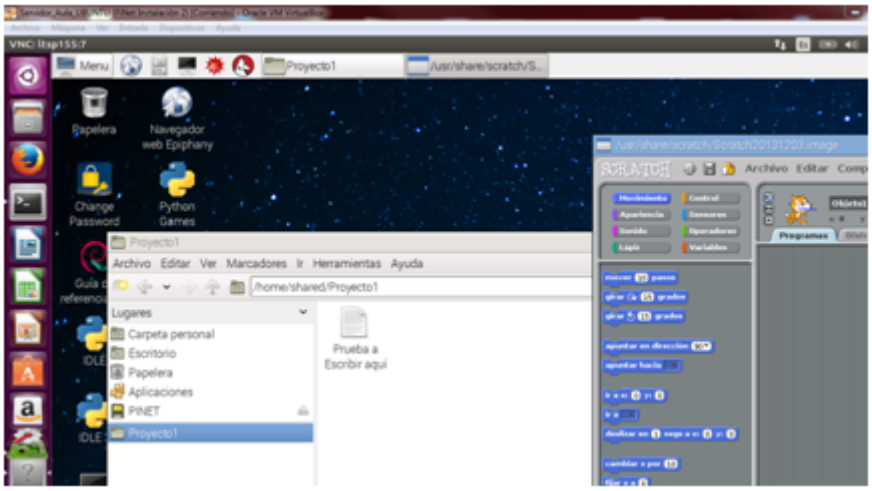
Nos aparecerá en la lista de ususarios y debermeos seleccionarlo. **Desde aplicaciones de Ubuntu** lo lanzamos



y en la ventana principal del programa podremos ver ya los equipos que ha detectado y que podremos pasar a utilizar, observar, "espiar"....



Así es como nos muestra la pantalla de uno de nuestros alumnos.



Para finalizar, cabe destacar que EPOPTES ofrece muchas funcionalidades; demostraciones en tiempo real de la realización de alguna tarea, mostar videos en la pantalla de nuestros alumnos, tomar el control de los equipos, bloquear - desbloquear sus pantallas, en definitiva nos permite dar una clase en nuestra aula de informática sin tener que ir equipo por equipo observando y ayudando a nuestro alumnado.

Copyright 2026

El uso de este programa supera los objetivos de este curso pero aquí os dejo uno de mis manuales favoritos para aprender las principales funcionalidades de este programa.Obviamente toda la parte de instalación que marca el tutorial no debemos hacerla puesto que acabamos de realizarla.

https://lihuen.linti.unlp.edu.ar/index.php/Epoptes

***/ TODAS LAS IMÁGENES UTILIZADAS EN EL PRESENTE MÓDULO HAN SIDO ELABORADAS POR EL AUTOR DEL CURSO.**

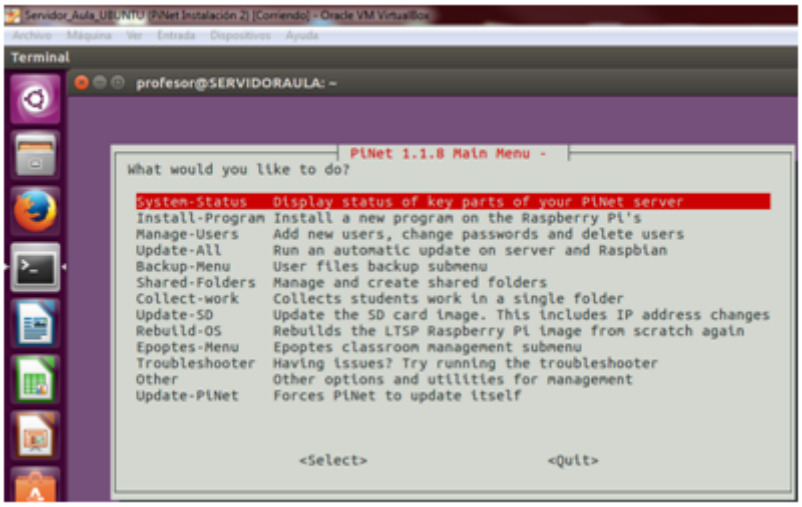
Copyright 2026

7. Otras opciones del Menú.

Comenzamos la última de las secciones en las que se divide el presente curso, esperamos quehasta ahora vaya todo bien y os este gustando el proyecto.

En este módulo, pretendemos explicar aquellas partes del Menú de configuración de PiNet que apenas usaremos o quizá sí, pero que no tienen un uso tan específico como las anteriores.

Arrancamos PiNet y vamos a detallar cada uno de las opciones del menú.



***/ System - Status:**

Nos ofrece de un rápido vistazo información sobre el esta del sistema:

Número de la versión.

Ip del servidor

Fecha de instalación de la versión.

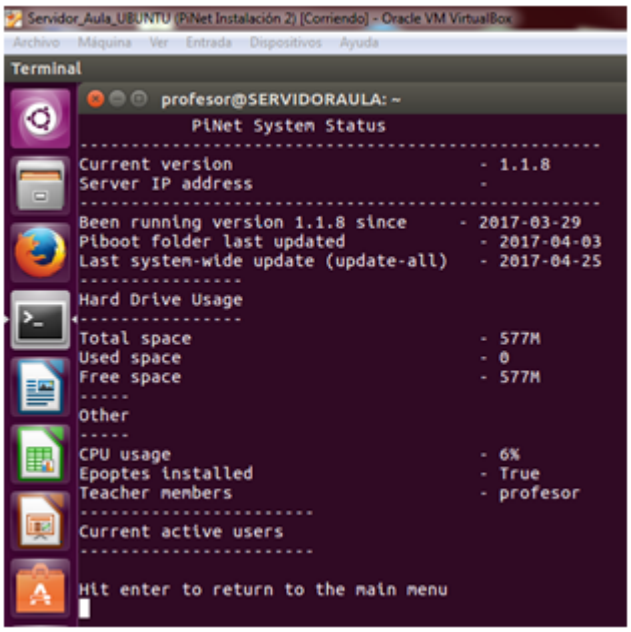
Fecha de la última actualización de PiBoot (sistema para las SD)

Fecha de la última actualización de todo el Sistema PiNet + Piboot

Uso del Disco Duro: Estos valores dependerán si hemos expandido o no las SD tras la instalación de Piboot.

Miembros del grupo Teachers

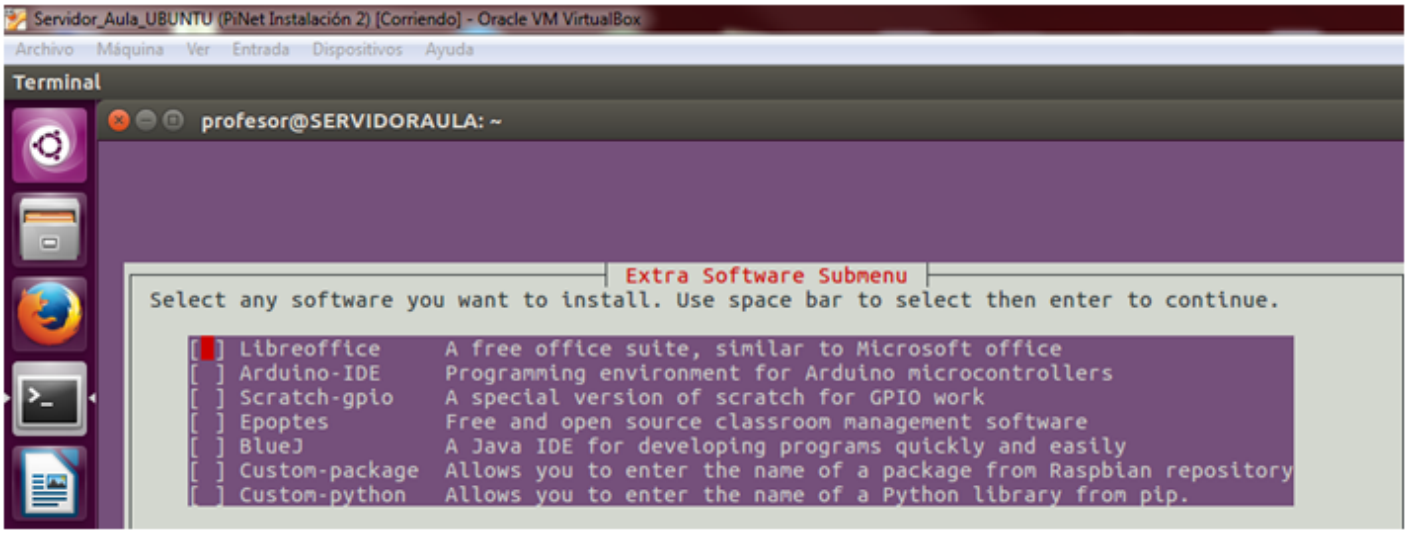
Usuarios activos en el momento.



Pulsamos Enter y Volvemos al Menú principal.

***/ Install - Program:**

Ya hemos hablado de él en el módulo anterior e igual que instalamos epoptes, podemos instalar el resto de programas de la lista.



Brevemente:

- Libreoffice: El equivalente a Microsoft Office, ya incluído en las Raspberrys, pero a lo mejor no en Ubuntu (lo normal es que sí)

- Arduino – IDE: Este programa se llama IDE, que significa "Integrated Development Environment" ("Entorno de Desarrollo Integrado"). Este IDE estará instalado en nuestro PC, es un entorno muy sencillo de usar y en él escribiremos el programa que queramos que el Arduino ejecute. Una vez escrito, lo cargaremos a través del USB y Arduino comenzará a trabajar de forma autónoma. Podemos saber más sobre él en el siguiente enlace.

https://openwebinars.net/blog/tutorial-arduino-ide-arduino/

- **Scratch – Gpio**: La adaptación de Scratch para el manejo del puerto gpio de nuestras Raspberrys. Aquí podeis documentaros más sobre su uso.

http://pitando.net/2015/09/10/scratchgpio-control-de-prototipos-con-scratch/

- Epoptes: Instalado en el módulo anterior.

- BlueJ: Adaptación del lenguaje Java para entornos educativos. Se basa en la programación orientada a objetos.

https://es.wikipedia.org/wiki/BlueJ

- Custom- package y Custom – Python: Para ambos sistemas, permite la creación de una carpeta compartida que actuará como repositorio de nuestro sistema.

***/ Manejo de Usuarios:**

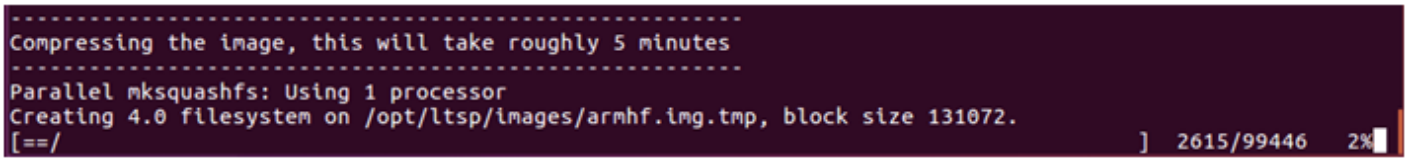
También explicado en módulos anteriores.

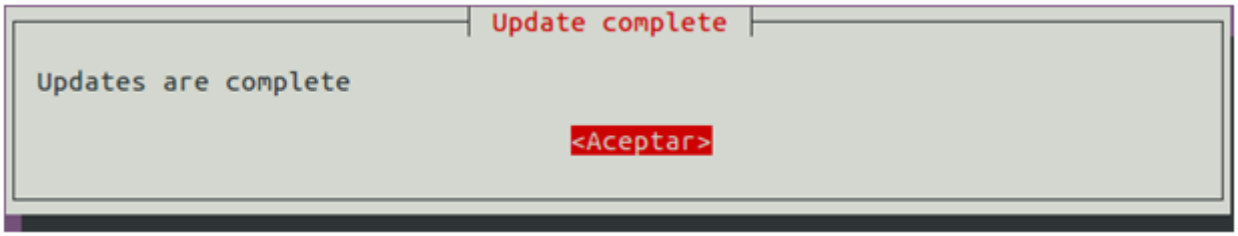
***/ Update - all:**

Realiza una actualización de todo el sistema del Servidor y de Raspbian. Cuesta bastante tiempo y conviene realizarlo cada pocos meses, sobre todo porque nos actualiza el software de todos los equipos.

Como cada vez que cambiamos algo, nos comprimirá la imagen del sistema.

Cuando finalice nos aparecerá un mensaje como este:





Nota:

****** Si al iniciar las Raspberry, tuvieramos algún problema, conviene copiar de nuevo en la SD el contenido de la carpeta Piboot, puesto que esta actualización puede haber cambiado algo importante de Raspbian.**

***/ Backup – menu:**

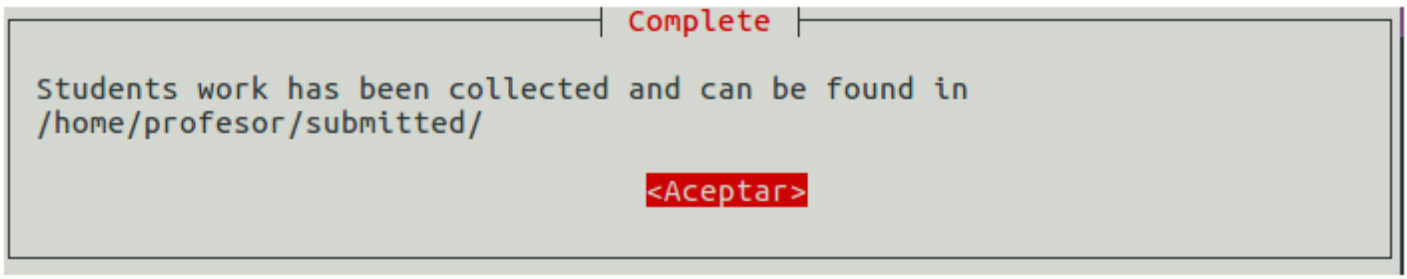
Usa archivos que tuvieramos guardados para recuperar el sistema de un fallo o una especia de vuelta a un estado anterior. Aún está en fase Beta, por lo que conviene crear otro tipo de respaldo por si acaso.

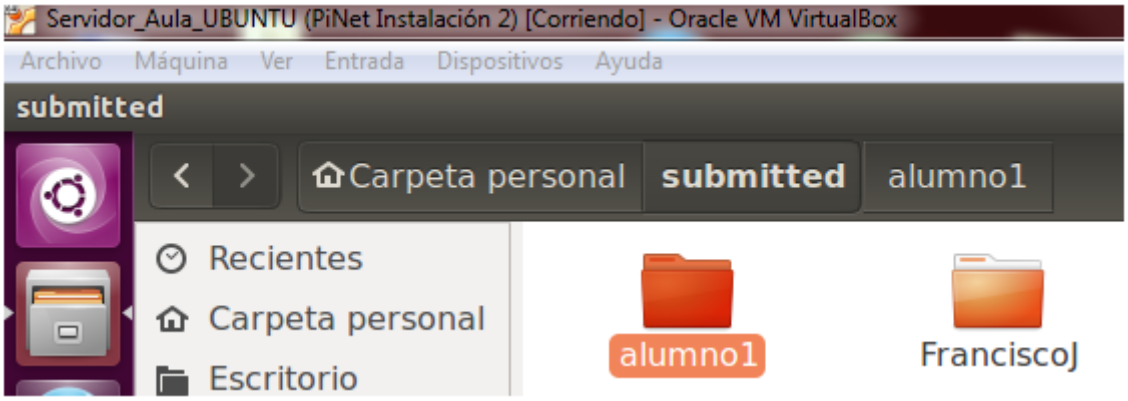
***/ Shared – folders: (carpetas compartidas):**

También estudiadas en módulos anteriores.

***/ Collect – Works:**

Permite recopilar trabajos de nuestros alumnos al momento es una sola carpeta, con lo que facilita mucho la tarea de recuperar todas las tareas que queramos “recoger” de nuestros alumnos. Solamente debemos entrar y poner el nombre del usuario que va a "recoger" los trabajos, aceptaremos el mensaje de aviso y veremos en la carpeta del profesor, una carpeta por alumno con el contenido coleecionado.



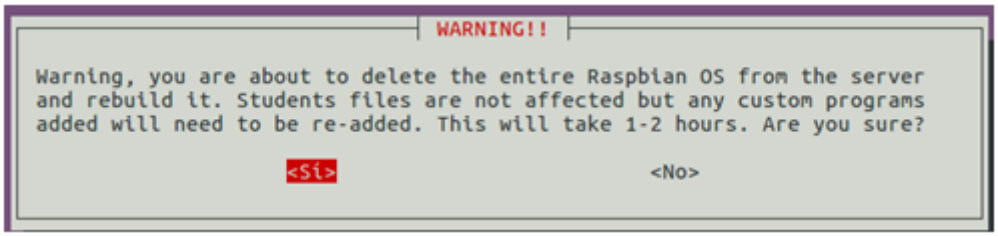


***/ Update – SD:**

Actualiza la imagen de Piboot para nuestras raspberrys, eso es necesario cuando se produzca un cambio en la IP del servidor o cuando queramos reinstalar los equipos de nuestros alumnos.

***/Rebuild – OS:**

Reconstruye el servidor LTSP de Linux soporte de Pinet. No es una opción muy utilizada ya que supone casi el completo formateo de Raspbian del servidor. Los archivos de los alumnos no deberían de ser afectados, pero si alguno de los programas.

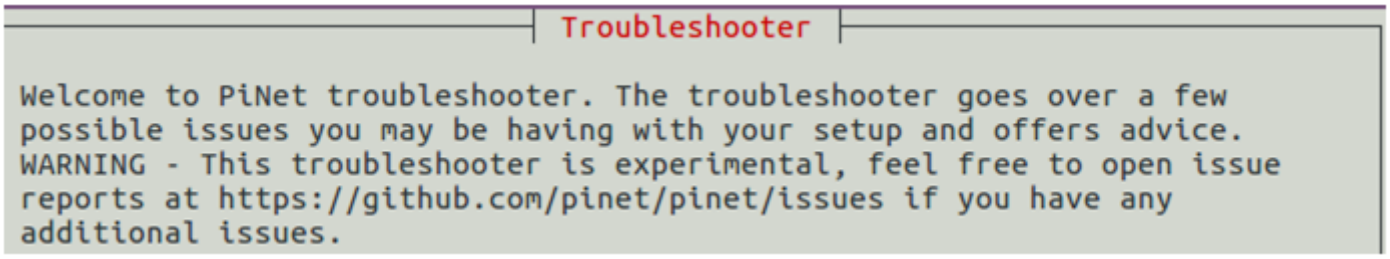


***/ Epoptes – menú:**

Visto en tareas anteriores arranca el programa epoptes.

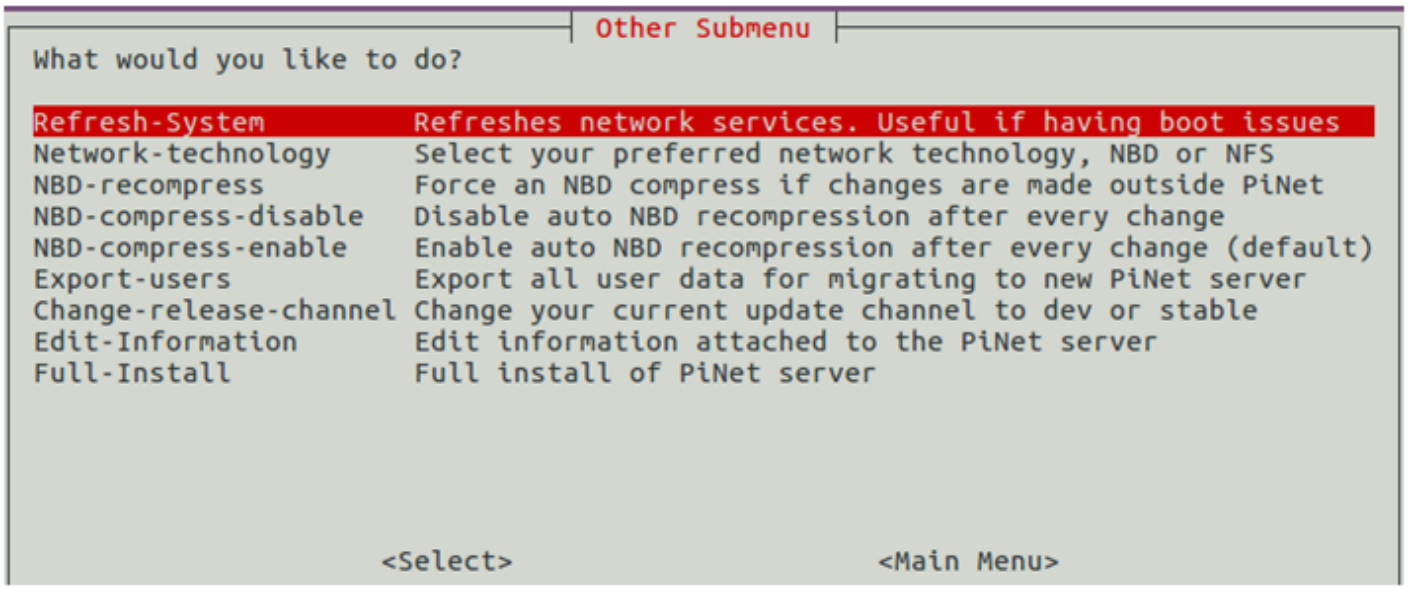
***/Troubleshooter:**

Es una especie de solucionador de problemas, que mediante una serie de preguntas nos puede ayudar a resolver algún pequeño problema de configuración que tuvieramos, con todo, nos acaba mandando a la página oficial de Pinet.



***/ Other:**

Otro tipo de configuraciones no accesibles desde el menú normal; las vemos:



Destacamos por su uso la primera, ya que es la que nos permite “refrescar” el sistema cuando hemos realizado algún cambio significativo en nuestra red.

El resto de opciones no es aconsejable cambiarlas para usuarios no avanzados.

***/ Update –Pinet:**

Fuerza el sistema a una total actualización, es recomendable cuando el propio sistema nos avise de que se ha publicado una nueva versión y para solucionar algún fallo de configuración inicial.

Y hasta aquí el módulo con más teoría de todo el curso, os invitamos a realizar la evaluación de la Unidad que es muy sencilla.

***/ TODAS LAS IMÁGENES UTILIZADAS EN EL PRESENTE MÓDULO HAN SIDO ELABORADAS POR EL AUTOR DEL CURSO.**

Copyright 2026

8. Software educativo incluído en la Raspberry Pi

Este módulo concluye el curso a modo de Epílogo, comentando alguna de las características, funciones y usos de los programas que podremos utilizar con nuestros alumnos en sus Raspberry´s, Realizar tareas con ellos es bastante sencillo y motivador para los alumnos. Esperamos que os gusten y os animéis a utilizarlas.



SCRACTCH : Es un lenguaje de programación visual, especialmente diseñado para que niños y jóvenes se inicien en tareas sencillas de programación. Gratuito, con multitud de proyectos accesibles en Youtube y en la propia página del desarrollador, en el que además de ejemplos, podremos acceder a áreas de consulta de datos. Por último cabe destacar que se puede utilizar también con recursos robóticos. Aquí os dejo un enlace en el que podréis saber más.

https://scratch.mit.edu/

MINECRAFT PI: Es un videojuego que está teniendo mucho éxito entre los chicos y chicas menores de 20 años. Con dos modos del mismo, supervivencia y creativo, nos ofrece un mundo infinito en el que poder desarrollar nuestras capacidades creativas, la toma de decisiones y la imaginación.

Quizá este sea el artículo donde mejor se explica cómo podemos adaptar el juego a nuestro alumnado y sacar el máximo partido de él.

http://www.educaciontrespuntocero.com/noticias/que-es-minecraft-educacion/32274.html

IDLE Y PYTHON: Aquí ya estamos hablando de programas mucho menos enfocados a jóvenes. Python es el propio lenguaje de programación, mientras que IDLE es el intérprete que nos ayudará en nuestra tarea de programar.

https://fabricadigital.org/2016/01/hola-pi-empezando-a-programar-en-raspberry-pi-con-python/

SONIC PI: Sonic Pi es una herramienta diseñada y creada por Sam Aaron y el equipo de Sonic Pi. Durante el proceso se estuvo muy en contacto con el profesorado de música y ciencias de la computación, y el resultado esuna herramienta de educación que abarcaesos dos campos a la vez. Es decir, **es una herramienta especialmente atractiva para aquellos niños**(o no tan niños) **interesados en la música y que quieran profundizar en ella, relacionándola con las matemáticas y la tecnología al mismo tiempo**.

http://pitando.net/2015/07/24/sonic-pi/

MATHEMATICA: Mathematica es un paquete de desarrollo para aplicaciones de tipo general en las que los aspectos de desarrollo matemático, algebraico, numérico, simbólico y gráfico juegan un papel preponderante. Mathematica está diseñado para tratar tanto cálculos matemáticos, así como aquellos cálculos numéricos tradicionalmente llevados a cabo en FORTRAN. La mayoría de funciones especiales de la física matemática y funciones matriciales, ya están construidas en Mathematica. Además, tiene una extensa capacidad gráfica para la visualización de los resultados de los cálculos.

www.xtec.cat/~fgomez/apmath.html

RASPBIAN:

Es el Propio sistema operativo del equipo de nuestros alumnos y con el podemos sacar todo el partido a las “Pi´s” aquí os dejamos algunos proyectos realizados mediante Raspberry Pi que creemos que son interesantes os invitamos a leerlos y por supuesto a crear los vuestros propios.

http://es.gizmodo.com/10-proyectos-de-raspberry-pi-ideales-para-principiantes-1791255833

https://www.xataka.com/accesorios/las-13-mejores-ideas-que-hemos-encontrado-hechas-con-raspberry-pi

Como ya comentábamos al principio del curso, este tipo de sistema Open Source favorece el aprendizaje colaborativo y la compartición de proyectos, dudas, sugerencias, ideas, etc.

Aquí tenéis el enlace al foro de la comunidad hispana de Raspberry Pi.

http://www.raspberrypi-spanish.es/foro/index.php?sid=3be38e5dd11162830ecfe1ad680f1e73

Créditos

Autor: Fernando Gasca Andreu

{% include "git+https://github.com/catedu/faq-aularagon.git/imagenes_creditos.md" %}

Copyright 2026