

alibabax



ÍNDICE

1. Qué es Raspberry?
 - 1.1. Cómo Funciona una Raspberry?
 - 1.2 Dónde comprar una Raspberry?
 - 1.3 Posibles usos con Raspberry?
 - 1.4 Preparación de raspberry

2. Configuración Sistema operativo.
3. Configuración wifi
4. Scripts
5. Alarma
6. Correo electrónico Caja
7. configuración mail postfix
8. puerta
9. preparación caja

Alí BaBox

Presentación

¿Alguna vez has notado que te falta dinero? Pues nosotros te damos la solución!

Esta no es una caja fuerte cualquiera, es una caja hecha para descubrir a quien quiera abrirla. Desde siempre ha habido casos de robos, cajas fuertes muy seguras que han sido forzadas y robadas sin saber quién ha sido, robos en casas, robos en tiendas con grabaciones en las que es imposible reconocer a nadie. Esta caja es la solución.

Al aparentar ser fácil de abrir(y lo es) el ladrón se confía, y comete errores. Lo que no sabe el ladrón, es que al forzar la caja una foto de su cara será subida a internet en segundos. Por mucho que robe, será fácil saber quién ha sido, denunciarlo y detenerlo.

Explicación

Alí BaBox es más una caja segura que una caja fuerte. El punto importante de esta caja no es la dificultad para abrirla, sino más bien la posibilidad de saber quién te ha robado o ha intentado robarte.

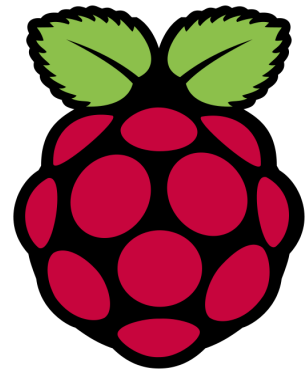
El funcionamiento de esta caja es sencillo. Tendrás una llave para abrir la caja y, en caso de que alguien intente forzarla, sonará una alarma que será una canción a toda potencia que “aturdirá” al ladrón, el cual, estará unos segundos sin saber que pasa. Al mismo tiempo que suena la alarma, una cámara fotografiará al ladrón y enviará dicha foto por mail.

Qué es Raspberry?

Raspberry es un ordenador pequeño diseñado en Estados Unidos por la fundación Raspberry pi para fomentar la enseñanza de las ciencias de computación en las escuelas.

La fundación Raspberry pi no expresa si el hardware es libre o no aunque si nos vamos a la página oficial al apartado de preguntas y respuestas, explican que tienen contratos de distribución y venta con dos empresas, pero al mismo tiempo cualquiera de este mundo puede convertirse en revendedor o redistribuidor de las tarjetas RaspBerry Pi, por lo que se entiende que es un producto con propiedad registrada pero de uso libre. De esa forma mantienen el control de la plataforma pero permitiendo su uso libre tanto a nivel educativo como particular. Tampoco deja claro si es posible utilizarlo a nivel empresarial u obtener beneficios con su uso, asunto que se debe consultar con la fundación.

La gran virtud de raspberry es su polivalencia ya que permite la creación de una infinidad de proyectos ya sean escolares o privados.



Cómo Funciona una Raspberry?

Para que funcione, basta con que añadamos nosotros mismos un medio de almacenamiento (como por ejemplo una tarjeta de memoria SD), conectarlo a la corriente gracias a cualquier cargador de tipo microUSB y, si lo deseamos, incorporar un chasis para que todo quede a buen recaudo y su apariencia sea más estética. Estos pueden ser desde cajas predeterminadas hasta una que fabriquemos nosotros mismos con grandes dosis de imaginación o incluso haciéndola con una impresora 3D.

La fundación de **Raspberry Pi** pone a disposición desde su página web Raspbian, una distribución de Linux basada en Debian, pero también podemos recurrir a muchas de las distribuciones específicas que la comunidad de usuarios ha desarrollado para diversos fines.

El sistema que nosotros usamos es Raspbian, tenemos más conocimientos de la distribución de Linux que la de Microsoft ya que en clase utilizamos Ubuntu y ambos sistemas son descendientes de Debian. Por eso los comandos son prácticamente los mismos y a la hora de trabajar nos es más fácil actuar de un modo o de otro.

En cuanto a conexión de red se refiere, podemos disponer de Ethernet para enchufar un cable RJ45 directamente al router o recurrir a adaptadores inalámbricos WiFi.

Nosotros no utilizaremos teclado ni ratón, haremos uso del protocolo ssh para gestionar nuestra Raspberry, una herramienta bastante útil para hacer una conexión a otro ordenador remoto en la cual puedes trabajar a gusto.

Donde comprar una Raspberry?

En Internet existen multitud de distribuidores que venden Raspberry Pi, y lo hacen tanto con la propia placa en sí como a través de útiles kits que incluyen cajas personalizadas o incluso otros accesorios como cámaras, adaptadores WiFi, tarjetas SD listas para utilizar y cables para la red eléctrica. Así, si no queremos dificultar el proceso podemos recurrir a ellos. Algunas de las webs donde puedes comprar una Raspberry Pi son www.raspberrypi.es, www.raspipc.es, Amazon y, por supuesto, muchas de las tiendas de informática tradicionales.

Posibles usos Con Raspberry

Dando un paseo por la red encontraremos decenas de usos para la Raspberry Pi. Uno de los más habituales es convertirlo en un media center para reproducir películas gracias a XBMC, convertirlo en un aparato que sustituya al Chromecast de Google o al Apple TV, que cumpla las funciones de reproductor musical centralizado e incluso hacer que ejerza de centro de control domótico en nuestro hogar, con proyectos como PiHome.

También podremos convertir a nuestra Raspberry Pi en un sistema de videovigilancia, crear un tanque con control remoto ([Raspberry Tank](#))... Como vemos, los usos son realmente imaginativos.

En nuestro caso teníamos desde un principio una idea bastante clara de lo que queríamos hacer así que la pusimos en marcha. Nuestra idea era hacer una caja fuerte distinta a las demás, una de la cual si intentas abrirla sin ser su propietario te hará una foto y la enviará por correo para facilitar tu identificación.

Preparación de la Raspberry

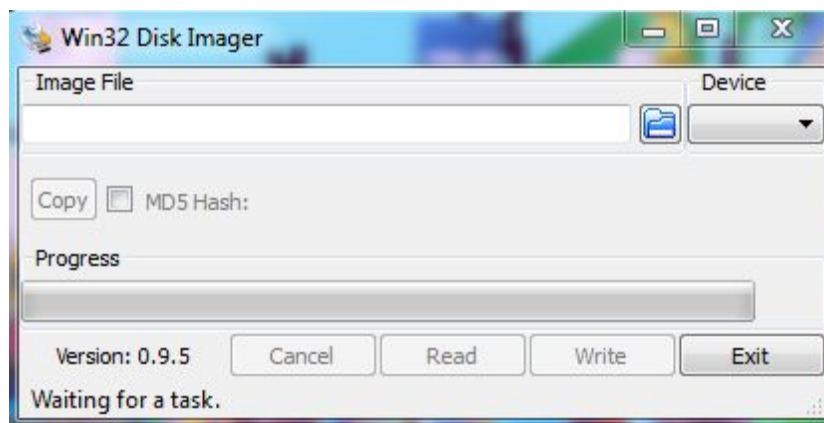
Introducir sistema operativo

Hemos escogido Raspbian como sistema operativo para este proyecto por su simplicidad y su proximidad al funcionamiento del sistema Ubuntu (dado que Ubuntu desciende del sistema Debian de la misma manera que Raspbian).

Para instalar Raspbian o cualquier otro sistema operativo en Raspberry debes seguir los pasos que facilitaremos a continuación.

Raspberry es un pequeño ordenador que no tiene disco duro, a no ser que conectes un disco duro USB. Normalmente funciona con una tarjeta MicroSD. Hay muchas maneras de instalar un sistema operativo en una Raspberry. El que nosotros hemos utilizado consiste en coger la SD y introducirla en un ordenador donde haremos lo siguiente:

1. Descargar e instalar el programa Win32 Disk Imager si lo haces desde Windows.
2. Abrir el programa y asegurarse de seleccionar el dispositivo extraíble correcto (normalmente :I).

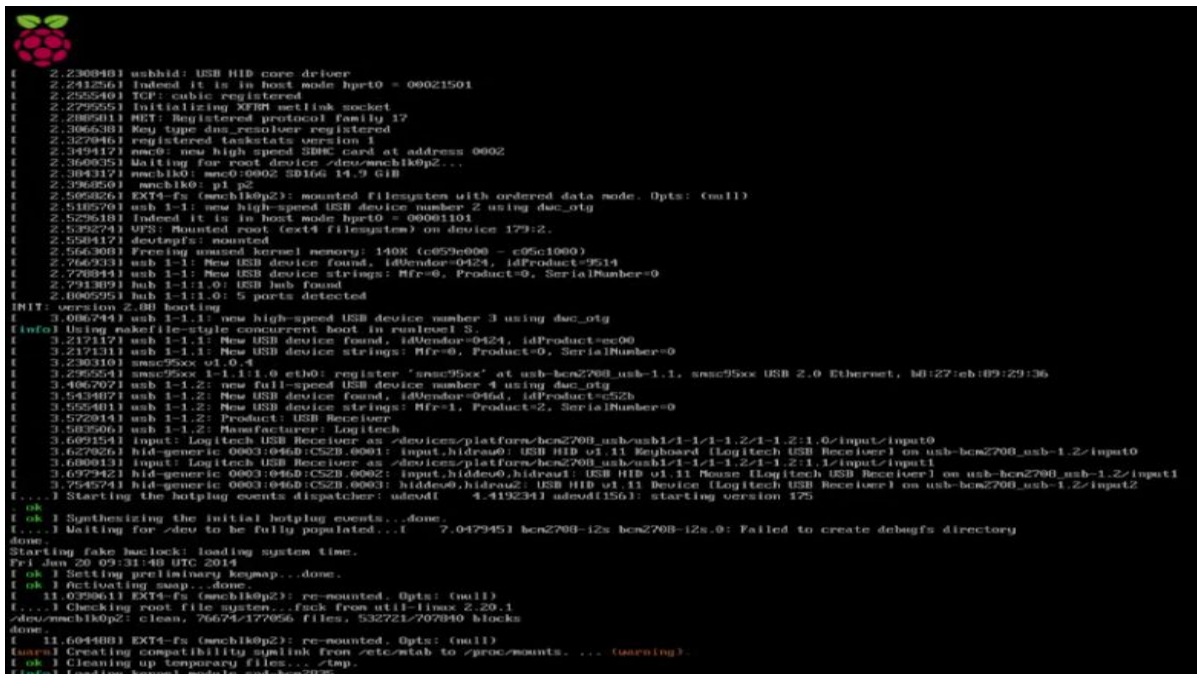


3. Esperar a que el proceso termine.
4. Sistema instalado, cuando el programa acabe su trabajo, sacamos la SD del ordenador y la volvemos a introducir en la Raspberry. Hecho todo esto ya tendremos el sistema en la Rasp y faltará configurarlo.

Configurar sistema operativo

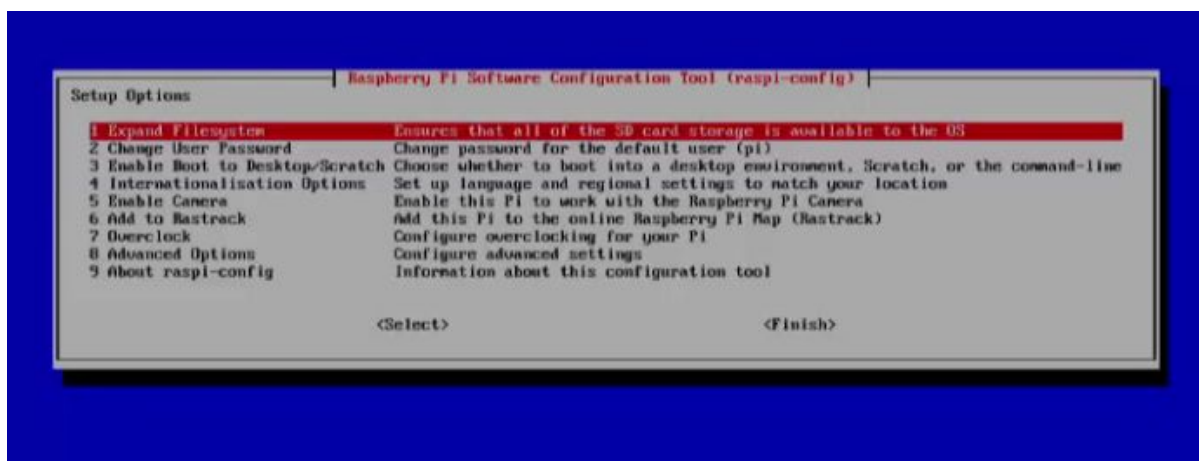
Para configurar el sistema operativo Raspbian tendremos que hacer lo siguiente:

1- Una vez introducido el sistema y encendida la raspberry nos saldrá una pantalla en negro donde nuestra raspberry comprueba los componentes antes de iniciar la configuración del sistema operativo raspbian.



```
2.2308481 usbbid: USB HID core driver
2.2412561 Indeed it is in host mode hprt0 = 00021501
2.2525540 TCP: cubic registered
2.2729253 Initializing ZIFC netlink socket
2.2885811 MPT: Registered protocol Family 17
2.3065383 Key type dns_resolver registered
2.3279946 registered taskstate version 1
2.3494171 mmc0: new high speed SDHC card at address 0002
2.3609351 Waiting for root device /dev/mmcblkp2...
2.3843171 mmcblk0: mmc0:0002:SD16G 14.9 GiB
2.3962501 mmcblk0: p1, p2
2.5058261 EXT4-Fs (mmcblk0p2): mounted filesystem with ordered data mode. Opts: (null)
2.5105781 usb 1-1: new high-speed USB device number 2 using duc_otg
2.5296181 Indeed it is in host mode hprt0 = 00001101
2.5392741 UFS: Mounted root (ext4 filesystem) on device 179:2.
2.5504171 devtmpfs: mounted
2.5663083 Freeing unused kernel memory: 140K (c059e000 - c05c1000)
2.7669331 usb 1-1: New USB device found, idVendor=0424, idProduct=9514
2.7708441 usb 1-1: New USB device strings: Mfr=0, Product=0, SerialNumber=0
2.7913891 hub 1-1:1.0: USB hub found
2.8005951 hub 1-1:1.0: 5 ports detected
INIT: version 2.88 booting
3.0867941 usb 1-1:1: new high-speed USB device number 3 using duc_otg
[info] Using rockfile-style concurrent boot in runitrcd S.
3.217171 usb 1-1:1: New USB device found, idVendor=0424, idProduct=ec00
3.217131 usb 1-1:1: New USB device strings: Mfr=0, Product=0, SerialNumber=0
3.2303191 smsc95xx v1.0.4
3.2955541 smsc95xx 1-1.1.0 eth0: register 'smc95xx' at usb-bcm2700_usb-1.1, smc95xx USB 2.0 Ethernet, b0:27:eb:09:29:36
3.4067071 usb 1-1:2: new full-speed USB device number 4 using duc_otg
3.4549071 usb 1-1:2: New USB device found, idVendor=046d, idProduct=c52b
3.5554811 usb 1-1:2: New USB device strings: Mfr=1, Product=2, SerialNumber=0
3.5729141 usb 1-1:2: Product: USB Receiver
3.5832061 usb 1-1:2: Manufacturer: Logitech
3.6091541 input: Logitech USB Receiver as /devices/platform/bcm2700_usb/usb1/1-1/1-1.2/1-1.2:1.0/input/input0
3.6270261 hid-generic 0003:046d:C52B.0001: input,hidraw0: USB HID v1.11 Keyboard [Logitech USB Receiver] on usb-bcm2700_usb-1.2/input0
3.6809131 input: Logitech USB Receiver as /devices/platform/bcm2700_usb/usb1/1-1/1-1.2/1-1.2:1.1/input/input1
3.6979421 hid-generic 0003:046d:C52B.0002: input,hidraw1: USB HID v1.11 Mouse [Logitech USB Receiver] on usb-bcm2700_usb-1.2/input1
3.7545274 hid-generic 0003:046d:C52B.0003: hidraw,hidraw2: USB HID v1.11 Device [Logitech USB Receiver] on usb-bcm2700_usb-1.2/input2
[....] Starting the hotplug events dispatcher: udevd[ 4.419234] udevd[115]: starting version 175
ok
ok [ Synthesizing the initial hotplug events...done.
[....] Making for /dev to be fully populated...[ 7.047945] bcm2700-12s: bcm2700-12s.0: Failed to create debugfs directory
done.
Starting fake huclock: loading system time.
Fri Jan 20 09:31:40 UTC 2014
ok [ Setting preliminary kmap...done.
ok [ Activating swap...done.
[ 11.039011 EXT4-Fs (mmcblk0p2): re-mounted. Opts: (null)
[....] Checking root file system...[ack from ntfs-11ms 2.20.1
/dev/mmcblk0p2: clean, 76674/177096 files, 53272/207040 blocks
done.
[ 11.6044801 EXT4-Fs (mmcblk0p2): re-mounted. Opts: (null)
[done] Creating compatibility symlinks from etc/stab to /proc/mounts. ... [warning]
ok [ Cleaning up temporary files.../tmp
[info] Loading kernel module snd-bcm2035
```

2- Seguidamente nos saldrá un cuadrado con todas las opciones para configurar nuestro raspbian correctamente.

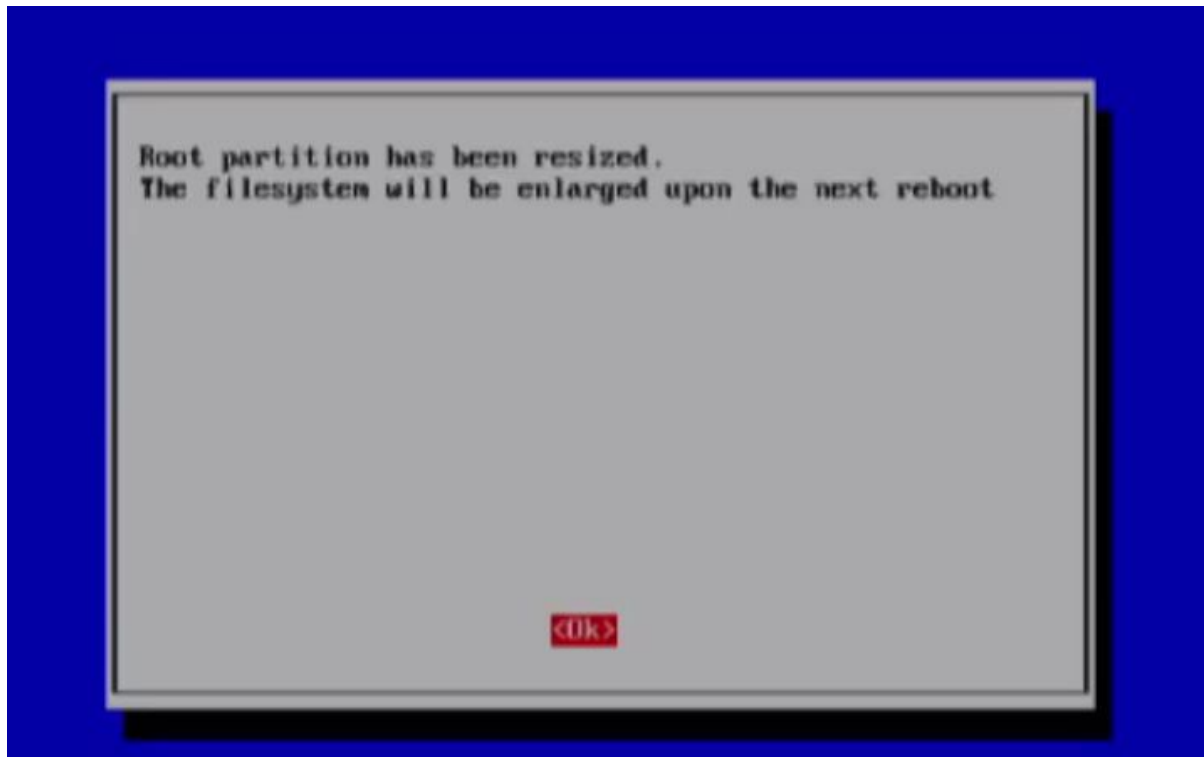


3- Es muy importante que utilicemos la opción Expand filesystem que es para expandir el sistema de archivos ya que por defecto este sistema solo formatea y utiliza 2 GB de espacio

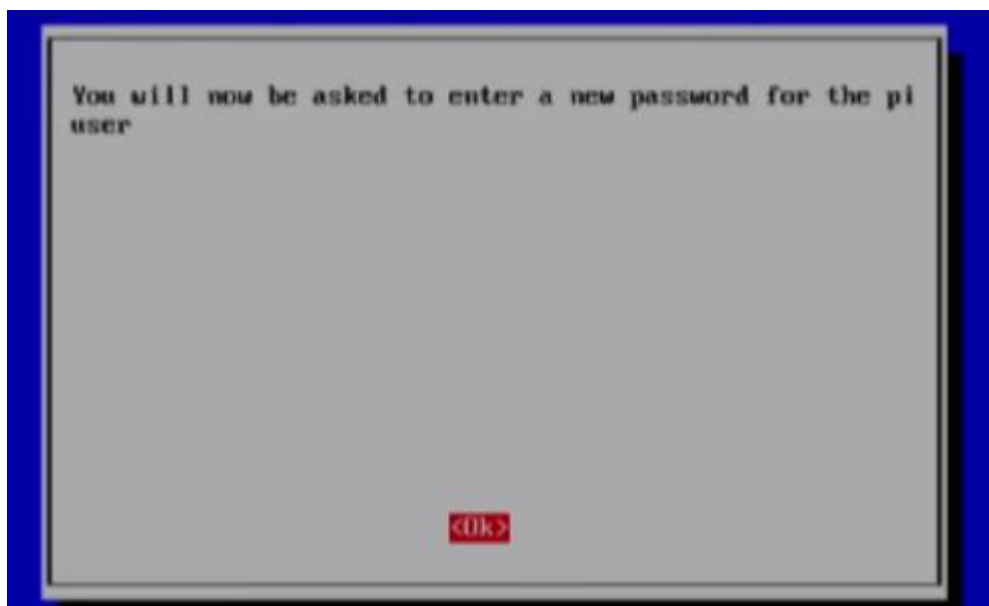
debido a que está pensado para utilizarse con pocos recursos. En caso de que tengamos poca capacidad de almacenaje está bien, pero si tenemos algo más es necesario hacer esto, sino nos quedaremos sin espacio rápidamente.



Este es el resultado que nos tiene que salir en pantalla una vez elegido esa opción.



5- A continuación nos vamos a ir a la opción dos vamos a cambiar el password del usuario raspbian por defecto vendrá el usuario pi y la contraseña raspberry para temas de seguridad es fundamental que como mínimo la contraseña se cambie una vez.



debajo de ese recuadro nos saldrá una parte en negro para que nosotros podamos poner la contraseña que queramos.

6-ahora vamos a la opción para activar la interfaz gráfica pero por defecto vendrá la interfaz de consola es decir por comandos.

7- seguidamente vamos a configurar el huso horario, seleccionaremos nuestro continente en nuestro caso pondré Europa y pondré de capital Madrid.

8- a continuación elegiremos la opción 3 para cambiar la configuración del teclado, le decimos que coja el teclado genérico si nuestro teclado esta en la lista le daremos a ese, como podemos observar solo sale el idioma en ingles si iniciamos así tendremos un problema con nuestro sistema operativo porque no sabremos las teclas que son, así que pulsaremos en la opción otros y cogemos la opción correspondiente a nuestro idioma y buscamos spanish.

9- una vez acabado esto dejaremos la configuración por defecto, le diremos que no queremos ningún atajo de teclado, nos preguntará si queremos configurar teclas rápidas para cambiar de modo gráfico a modo comando le tendremos que dar a que no sino habrá muchos fallos y nos cargaremos el sistema operativo.

10- con este último paso ya prácticamente hemos acabado la configuración así que iremos a las opciones avanzadas por si queremos actualizar el sistema operativo mientras se reinicia sino lo dejaremos tal cual, una vez seguido los pasos le damos a finish.

Instalaciones/Acciones recomendables (opcional)

En este apartado explicaremos la instalación de programas que recomendamos instalar para seguir con este proyecto.

Editor de texto Vim

sudo apt-get install vim

sudo apt-get upgrade vim

De esta manera tendremos la última versión del editor Vim, que es muy útil para editar los archivos de configuración o editar cualquier tipo de archivo y nos aseguramos de que no tenga fallos.

Cambiar el idioma del teclado y del sistema

Para cambiar el idioma del sistema:

```
sudo dpkg-reconfigure locales
```

Una vez veamos la lista de idiomas, escogemos los que nos convengan, que en mi caso son todos los que empiezan por “es_ES...”. Después seleccionamos el idioma predeterminado, en mi caso “es_ES@euro”.

Para cambiar el idioma del teclado:

```
sudo dpkg-reconfigure keyboard-configuration
```

```
0
```

```
sudo raspi-config
```

Nota: En nuestra versión del SO (Raspbian jessie) no ha funcionado ninguno de los dos, pero si entras en la rasp haciendo ssh desde una máquina con un teclado bien configurado evitas el problema.

Nota: Por lo que he leído, en otras versiones si que funciona, así que no cejeis en intentarlo.

Servidor SSH

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

Recomiendo instalar un servidor ssh para poder acceder a la Raspberry desde diferentes maquinas, es util cuando quieres trabajar en una máquina que no puedes conectar con HDMI.

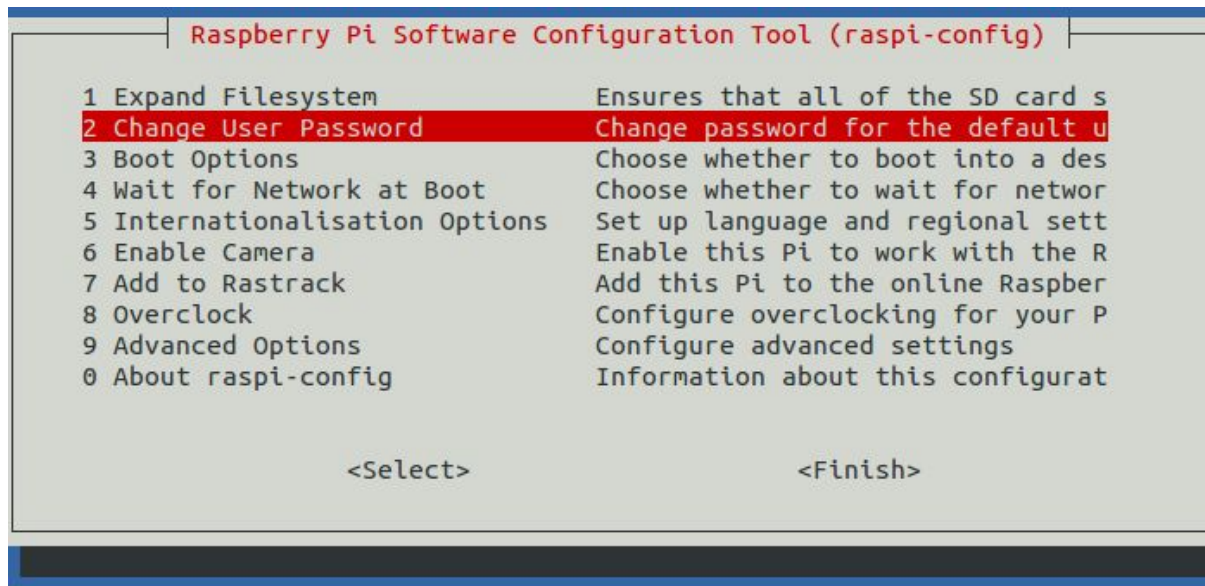
Cambiar contraseña del usuario Pi

Es recomendable por seguridad, cambiar la contraseña del usuario Pi.

Puedes hacerlo de manera sencilla entrando en la configuración con este comando

```
sudo raspi-config
```

Una vez dentro todo es bastante visual, seleccionamos la opción “Change user Password” y ponemos la contraseña que queramos.



Proceso del proyecto: Software

Red

Después de todo lo explicado, empezamos a trabajar en el proyecto. Lo primero que hemos hecho ha sido poner una IP estática en una tarjeta de red virtual a parte de la IP primaria que ya viene configurada por defecto (IP dinámica).

Lo hemos hecho editando el archivo `/etc/network/interfaces` y añadiendo estas líneas:

```
auto eth0:1
iface eth0:1 inet static
address 10.40.40.100
netmask 255.255.255.0
```

El objetivo de esta configuración es poder conectar por **ssh** sin importar la red en la que esté la caja, levantando una IP secundaria en el ordenador desde el cual te quieras conectar a la Raspberry, que a partir de ahora llamaremos cariñosamente Raspi, para abreviar.

Nota: Para levantar una IP secundaria en la máquina cliente (la máquina que quiere conectarse a la raspi) utilizaremos este comando

```
ifconfig eth0:1 10.40.40.X netmask 255.255.255.0 up
```

Configurar wifi

En algunos casos sería posible utilizar WiFi para las funciones de red, lo cual proporciona algo más de seguridad física a la caja dado que no podrían desconectar el cable para interrumpir el envío de la fotografía por ejemplo. Por ello aquí explicaremos como utilizar wifi en Raspbian.

Para poder utilizarlo necesitaremos un **Dongle WiFi**, ya que raspberry no tiene adaptador wifi de serie.



Una vez conectado el dongle ejecutamos esto para asegurarnos de que el software necesario está instalado y actualizado.

```
sudo apt-get update  
sudo apt-get install wpasupplicant
```

Restringimos los permisos del archivo *interfaces* para evitar que la clave sea revelada

```
sudo chmod 0600 /etc/network/interfaces
```

Ahora utilizaremos el comando *wpa_passphrase* para calcular el hash wpa-psk correcto para nuestra red

```
sudo wpa_passphrase ssid mi_contraseña
```

NOTA: El SSID de es el nombre de nuestra red.

Este comando nos dará un valor alfanumérico de 40 caracteres, que a la hora de configurar la red utilizaremos para establecer la contraseña.

Ahora tenemos que abrir el archivo *interfaces* y escribir lo siguiente:

```
sudo vi /etc/network/interfaces  
  
auto wlan0  
iface wlan0 inet dhcp  
wpa-ssid SSID  
wpa-psk ElHashQueTeHaDadoElComandoAnterior
```

Cierra el editor de texto guardando los cambios y reinicia la máquina.

Scripts

En esta sección explicaremos la función de cada script y como los hemos organizado, también pondremos aquí el código de los scripts.

La estructura de programación estará basada en un script principal que será el que controle el estado de la puerta. Este script estará enlazado en su mismo código a otros dos scripts secundarios que harán el resto de las funciones

Alarma

La caja disparara una alarma cuando la puerta se abra sin la llave.

Para realizar esta acción utilizaremos un altavoz conectado por USB a la raspi, un reproductor MP3 de terminal llamado **PLAY**, y una canción.

Para empezar tenemos que instalar el paquete multimedia **sox** en la Raspberry.

```
sudo apt-get install sox libsox-fmt-mp3
```

Con esto ya podremos utilizarlo, después de esto necesitamos tener la canción en la raspi. Podemos obtenerla de dos maneras, puedes descargarla directamente en la rasp, o descargarla en una máquina mayor y pasarla a la rasp mediante el comando **scp**. Nosotros hemos usado la última con este comando

```
scp Ruta/de/tu/alarma.mp3 pi@10.40.40.100:/home/pi/
```

Hecho todo esto, escribimos la orden dentro del script *puerta.sh* que lo veremos mas tarde.

play /ruta/de/la/cancion.mp3

Fotografiar y enviar por mail

Este script se ejecutará cuando se den las condiciones de que la puerta esté abierta y la llave no haya sido introducida. Su función es fotografiar a la persona que abra la caja y enviar la fotografía por email a una dirección de correo.

Para ejecutar este script utilizaremos varios recursos:

- Raspberry Pi Cam
- Comandos correspondientes a Raspberry Pi Cam como *raspistill*
- Comando *mail*

Antes de poder utilizar el script requerimos una serie de configuraciones previas.

Cuenta de correo electrónico de la caja

Este paso es opcional, ya que puedes configurar la rasp con tu propio email(lo veremos mas adelante) pero queda mejor si la caja tiene su propio correo.

Vamos a gmail.com y pinchamos en el enlace create account, una vez dentro nos saldra unos recuadros para poner el nombre y todo.

Crea tu cuenta de Google

Solo necesitas una cuenta

Accede a todos los servicios de Google con una cuenta gratuita.



Tu información siempre disponible

Cambia de dispositivo y continúa desde la última acción que hayas realizado.



Nombre

Nombre de usuario

@gmail.com

Contraseña

Confirma tu contraseña

Fecha de nacimiento

Sexo

Selecciona tu sexo

Teléfono móvil

+34

Tu dirección de correo electrónico actual

Demuéstranos que no eres un robot

rellenamos todos los campos necesarios para crearnos el correo, como ya hemos dicho antes el nombre será ali babox.

una vez que rellenemos todos esos campos nos saldrá una pestaña con las condiciones y le daremos a acepto.

Privacidad y condiciones

Si haces clic en **Aceptar** a continuación, significa que aceptas las [Condiciones de servicio](#) de Google.

Significa también que aceptas nuestra [Política de Privacidad](#), donde se explica cómo procesamos tu información, incluidos estos aspectos clave:

Datos que tratamos cuando usas Google

- Cuando usas servicios de Google para acciones como escribir un mensaje en Gmail o comentar un vídeo de YouTube, almacenamos la información que creas.
- O, por ejemplo, si buscas un restaurante en Google Maps o ves un vídeo en YouTube, procesamos la información sobre esa actividad (como el vídeo que has visto, los ID de dispositivo, las direcciones IP, los datos de cookies y la ubicación).

[CANCELAR](#) [ACEPTO](#)

*Acceder a aplica:
notificaciones y*

¡Te damos la bienvenida!

Tu nueva dirección de correo electrónico es alibabox11@gmail.com.

Gracias por crear una cuenta de Google. Te permitirá suscribirte a los canales de YouTube, mantener un chat de vídeo de forma gratuita, guardar tus lugares favoritos en Google Maps y mucho más.

[Ir a Gmail](#)



una vez aceptado las condiciones ya tendremos creado el correo de nuestra caja.

Conectar y habilitar cámara

Conectar

Conectar la cámara es bastante sencillo, pero hay que tener en cuenta que es un aparato muy sensible y que se estropea con facilidad, por lo que debemos tener mucho cuidado de no conectarla ni desconectarla de la Raspberry estando ésta conectada a la alimentación.

El conector de la cámara se llama Sunny, Raspberry tiene una clavija especialmente reservada para este dispositivo.

Para conectar la cámara primero levantaremos la sujeción que tiene la clavija para el conector como se ve en la foto:



Después introduciremos el conector Sunny de la cámara en la clavija con el plástico azul del final del conector mirando en dirección al conector RJ-45.

Después de esto solo queda bajar la clavija de manera que el conector quede bien sujeto y la cámara ya estará conectada.

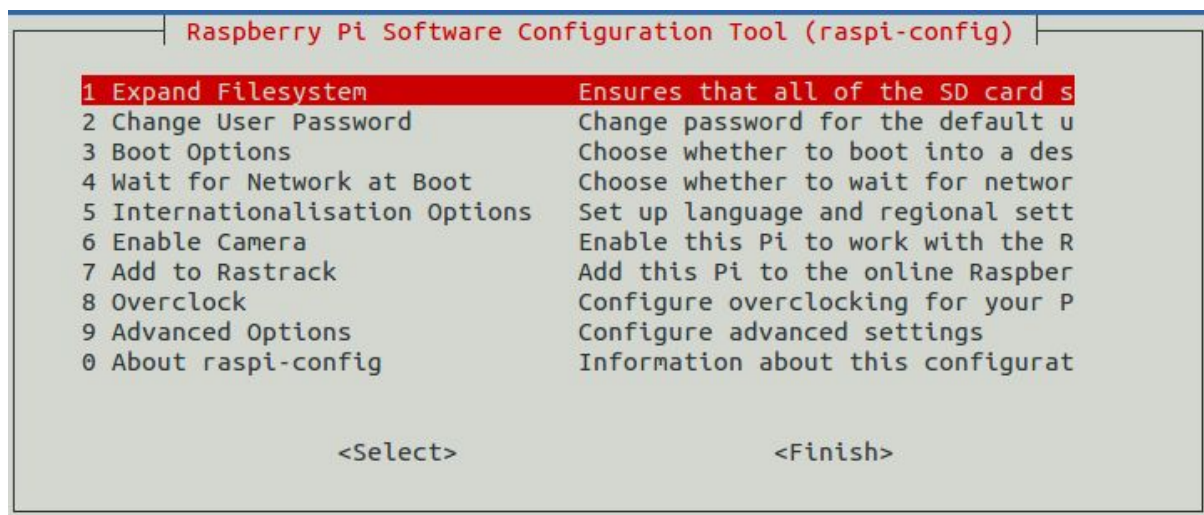


Habilitar

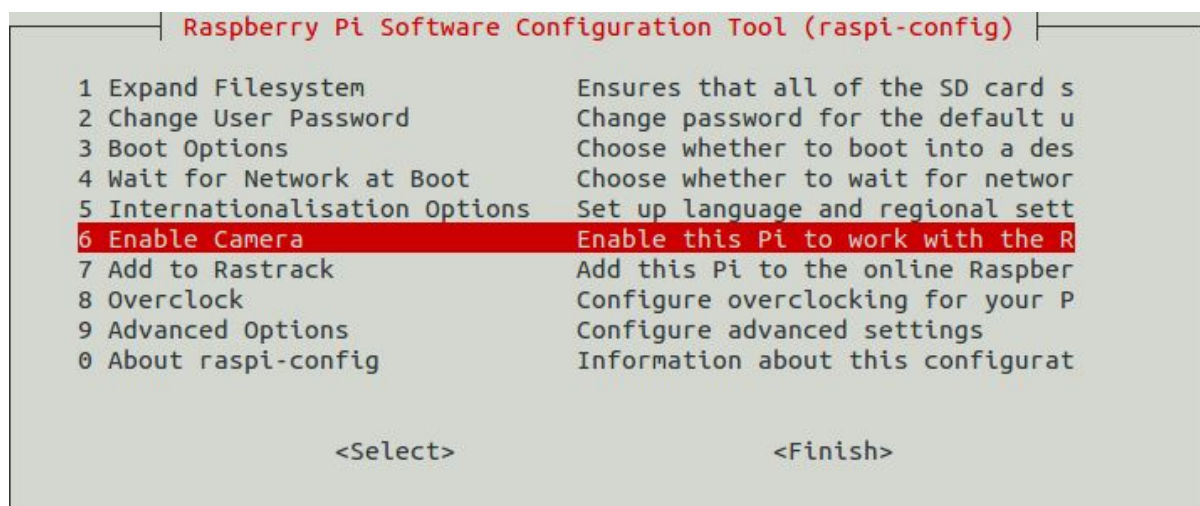
Una vez conectada la cámara, la encenderemos y procederemos a habilitarla. Para ello entraremos en la Raspberry y entraremos en su configuración con este comando:

sudo raspi-config

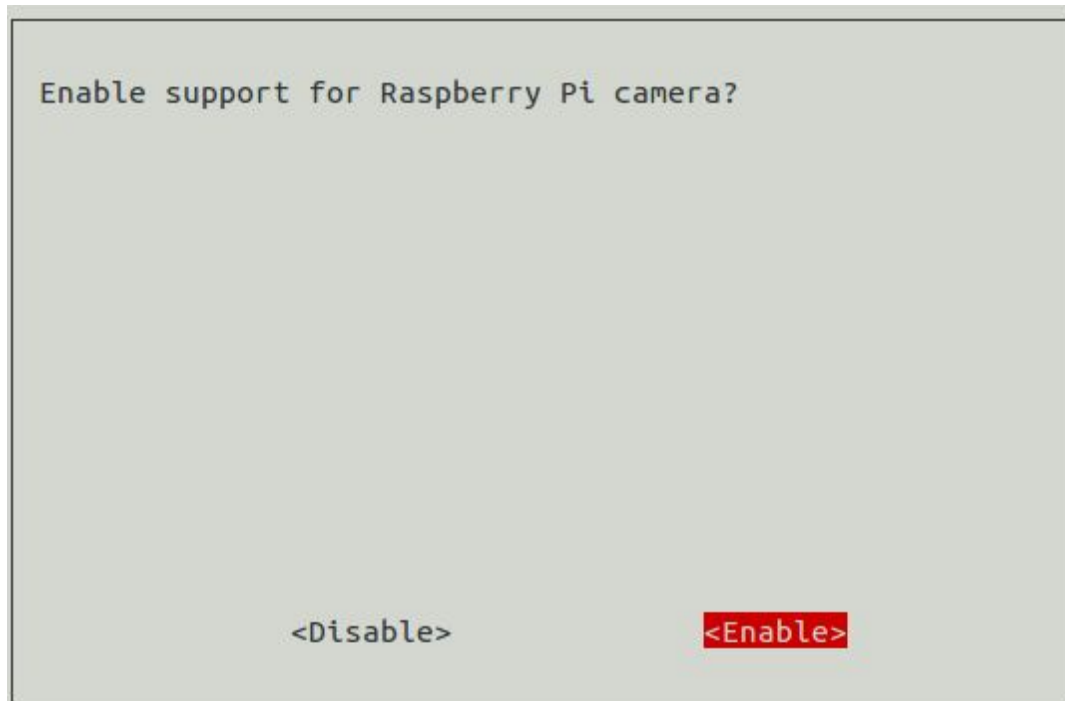
Entonces nos saldrá esta ventana:



Cuando estemos en este menú escogeremos la opción 6 “**Enable camera**”



Se abrirá otra ventana preguntándonos si queremos habilitar la cámara, seleccionaremos “enable”



Hecho esto saldremos del menú, nos preguntará si queremos reiniciar y le diremos que si.
La cámara está lista para usarse

Enviar correo

Para poder enviar correo electrónico desde nuestra Ali BaBox necesitaremos instalar y configurar los programas *mail*, *postfix* y *ssmtp*. Los tres son totalmente necesarios para poder enviar mensajes a través de gmail desde la Rasp.

Configurar *mail* y *postfix*

El comando Mail es una herramienta simple pero muy funcional a la hora de enviar correo electrónico desde la terminal

Se puede utilizar de forma interactiva, con *mail usuario@gmail.com* es decir vía terminal.

```
pi@raspberrypi:~ $ mail flopez@elpuig.xeill.net
Cc:
Subject: jelou
jelou esto es un test
```

Configuración para poder mandar mails a través de Gmail

Primer paso: instalar los paquetes necesarios para su uso estos paquetes son postfix, mailutils y libsasl. Para ello utilizaremos el siguiente comando:

```
sudo apt-get install postfix mailutils libsasl2-2 ca-certificates libsasl2-modules
```

Segundo paso: tendremos que entrar en el fichero de configuración de postfix para ello utilizaremos el comando:

```
vim /etc/postfix/main.cf
```

dentro del fichero de configuración de postfix tendremos que modificar la configuración de manera que quede así:

```
relayhost = [smtp.gmail.com]:587  
smtp_sasl_auth_enable = yes  
smtp_sasl_password_maps = hash:/etc/postfix/sasl_passwd  
smtp_sasl_security_options = noanonymous  
smtp_tls_CAfile = /etc/postfix/cacert.pem  
smtp_use_tls = yes
```

Tercer paso: entraremos en el fichero de configuración de postfix nuevamente pero esta vez en el de passwd para poner nuestro correo y contraseña que queramos utilizar para nuestra caja fuerte, para ello utilizaremos el comando

```
vim /etc/postfix/sasl_passwd
```

seguidamente una vez que estés dentro del fichero tendremos que insertar la siguiente línea con nuestro correo electrónico y nuestra contraseña.

```
[smtp.gmail.com]:587 USERNAME@gmail.com:PASSWORD
```

una vez hecho este paso tendremos que darle los permisos necesarios para poder usar el fichero sasl_passwd, para ello utilizaremos el comando;

```
sudo chmod 400 /etc/postfix/sasl_passwd  
sudo postmap /etc/postfix/sasl_passwd
```

Último paso: A continuación, tendremos que validar los certificados para evitar errores que puedan salir así que para ello utilizaremos el siguiente comando:

```
cat /etc/ssl/certs/thawte_Primary_Root_CA.pem | sudo tee -a /etc/postfix/cacert.pem
```

Para acabar tendremos que reiniciar nuestro servidor postfix para que todo quede correcto y ya habremos acabado, para ello utilizaremos el comando ;

```
sudo /etc/init.d/postfix reload
```

Una vez configurados los programas mail y postfix empezamos a tocar el programa **ssmtp**. Esta parte es muy importante ya que ssmtp será quien cifre nuestros mensajes para que nadie los bloquee.

Primero vamos a instalarlo y actualizarlo por si acaso.

```
sudo apt-get update  
sudo apt-get install ssmtp  
sudo apt-get upgrade ssmtp
```

Antes de seguir necesitamos configurar el programa, para ello entramos en su archivo de configuración

```
sudo vim /etc/ssmtp/ssmtp.conf
```

Modificamos las siguientes líneas del archivo de configuración de tal manera que quede así:

```
root=MyEmailAddress@gmail.com  
mailhub=smtp.gmail.com:587  
AuthUser=MyEmailAddress@gmail.com  
AuthPass=MyPassword  
UseTLS=YES  
UseSTARTTLS=YES  
rewriteDomain=gmail.com  
hostname=MyEmailAddress@gmail.com  
FromLineOverride=YES
```

En caso de que las líneas no están escritas introducelas.

Después de todo esto, es totalmente imprescindible activar el permiso para aplicaciones menos seguras, si no lo haces Google bloqueará todos los mensajes que intentes enviar.

Para hacer esto tienes que entrar en tu cuenta de Gmail (o en la que quieras usar), abrir el menú desplegable de aplicaciones de Google y pinchar en “Mi cuenta”

Cuando se abra la nueva ventana, buscamos la opción “Inicio de sesión y seguridad”



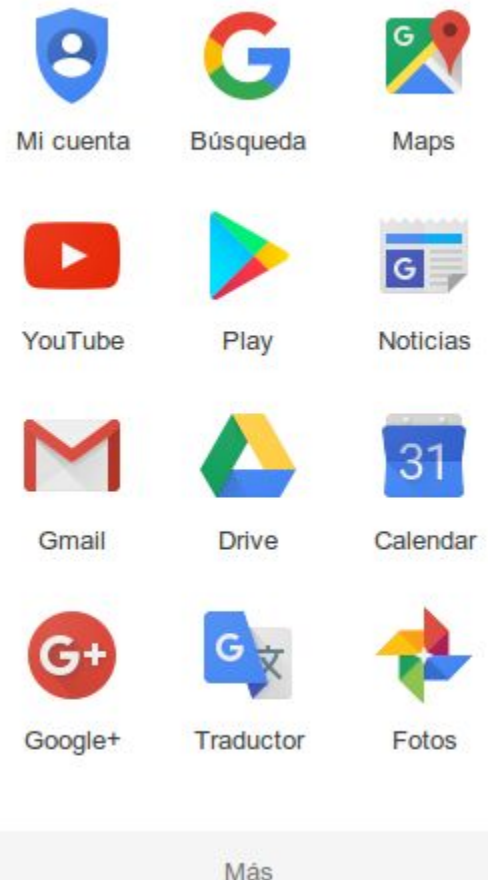
Inicio de sesión y seguridad >

Controla tu contraseña y la configuración de acceso a la cuenta.

[Inicio de sesión en Google](#)

[Notificaciones y actividad en tus dispositivos](#)

[Aplicaciones y sitios conectados a tu cuenta](#)



Dentro de esta página, bajamos hasta el final donde encontraremos la opción “Permitir el acceso de aplicaciones menos seguras”. Tenemos que asegurarnos de que está marcado el SÍ.

Permitir el acceso de aplicaciones menos seguras: SÍ

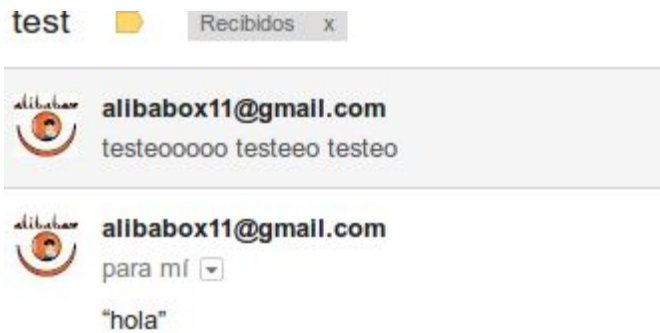


Algunas aplicaciones y dispositivos que no son de Google usan una tecnología de inicio de sesión menos segura, lo cual puede hacer que tu cuenta sea más vulnerable. Te recomendamos que desactives el acceso de estas aplicaciones. No obstante, puedes elegir usarlas, a pesar del riesgo que ello supone.

Probando

Para probar que el comando mail funciona ejecutaremos un comando simple y observaremos si llega a su destino

echo "hola" | mail -s "test" tucorreelectronico@gmail.com (o cualquier dirección)



Viendo que el comando mail funciona, vamos a probar el comando ssmtp.
Para probar el comando ssmtp utilizaremos este comando

ssmtp roger.mocholi@gmail.com
To:roger.mocholi@gmail.com
From:alibabox11@gmail.com
Subject:Test
hola esto es un test

Explicación: cuando ejecutas el comando resaltado en negrita, tienes que escribir el cuerpo del mensaje en texto plano, definiendo el origen el destinatario y el asunto tal como está ejemplificado.

```
pi@raspberrypi:~ $ ssmtp roger.mocholi@gmail.com
To:roger.mocholi@gmail.com
From:alibabox11@gmail.com
Subject:Test

hola esto es un test
```

Resultado



Para poder enviar un archivo adjunto con el comando mail escribiremos esto:

```
echo "he aquí una foto" | mail -s "Si, es una foto" -A /ruta/de/lafoto.jpg  
estestumail@ejemplo.com
```

Código del script

Después de todo este proceso de configuraciones e instalaciones, podemos por fin escribir el script, que es muy corto en comparación con todo lo que hay que hacer para que funcione.

Creamos un archivo llamado "foto.sh" y dentro escribimos lo siguiente:

```
#!/bin/bash  
  
date=$(date "+%d-%m-%Y_%H%M")  
  
raspistill -o $date.jpg  
  
echo "Contenido del mensaje" | mail -s "Asunto del mensaje" -A $date.jpg  
correo@electronico.com
```

Lo que hace este script es lo siguiente:

Con la primera línea (`date=...`) establece que la variable "date" tenga como valor el resultado del comando que hay entre paréntesis, es decir que el valor de "date" sería la fecha en formato DD-MM-YY seguido de la hora.

Ej.

```
date=10-05-2016_1320
```

La segunda línea lanza la orden de hacer una foto y nombrarla con el valor de la variable `date`.

La tercera envía un correo electrónico adjuntando la foto que acabamos de hacer a una cuenta de correo concreta.

Puerta

Este script es el más importante de todos, se encargará de detectar la disposición de la puerta (abierta o cerrada) y de ordenar en cada caso la ejecución del script correspondiente. El orden de acción es el siguiente:

Si la puerta está cerrada volver a ejecutar el script

Si la puerta está abierta con acceso permitido espera a que la puerta se cierre y entonces vuelve a ejecutar el script.

Si la puerta esta abierta con acceso denegado ejecuta el script “foto.sh” y reproduce la alarma

una vez que cumpla su función, la raspberry enviará un correo electrónico a tu gmail vinculado, con la imagen del presunto ladrón, la hora y el día de cuando hizo el atraco.

Detección de puerta

Para detectar si la puerta está abierta o cerrada utilizaremos un sensor magnético llamado “Reed switch” . Consiste en dos imanes que al separarse generan una respuesta distinta a cuando están juntos. Pondremos un imán en la puerta y otro en la caja.

Para poder conectar este sensor a la Raspberry hemos tenido que hacer un apaño, dado que necesitábamos un conector hembra.

Hemos cogido dos cables hembra-macho y hemos soldado los lados macho, de manera que queda un cable hembra-hembra.

Hemos conectado el sensor a pines GPIO genéricos, ya que este no necesita estar conectado a un pin de alimentación



En nuestro caso hemos conectado el sensor a los pines 13 y 14. Mucho cuidado con el número del pin, ya que la numeración a la hora de programar los pines es distinta. Para saber toda la información necesaria sobre los pines de la raspberry debes ejecutar el comando

gpio readall

Que te mostrará una tabla como esta con las diferentes denominaciones de cada pin y sus valores/estados.

```

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| BCM | wPi | Name | Mode | V | Physical | V | Mode | Name | wPi | BCM |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|      |      | 3.3v |      |   | 1 || 2 |      |      | 5v  |      |      |
| 2  | 8  | SDA.1 | IN  | 1 | 3 || 4 |      |      | 5V  |      |      |
| 3  | 9  | SCL.1 | IN  | 1 | 5 || 6 |      |      | 0v  |      |      |
| 4  | 7  | GPIO. 7 | IN  | 1 | 7 || 8 | 1 | ALT0 | TxD  | 15  | 14  |
|      |      | 0v  |      |   | 9 || 10 | 1 | ALT0 | RxD  | 16  | 15  |
| 17 | 0  | GPIO. 0 | IN  | 0 | 11 || 12 | 0 | IN  | GPIO. 1 | 1  | 18  |
| 27 | 2  | GPIO. 2 | IN  | 0 | 13 || 14 |   |      | 0v  |      |      |
| 22 | 3  | GPIO. 3 | IN  | 0 | 15 || 16 | 0 | IN  | GPIO. 4 | 4  | 23  |
|      |      | 3.3v |      |   | 17 || 18 | 0 | IN  | GPIO. 5 | 5  | 24  |
| 10 | 12 | MOSI | IN  | 0 | 19 || 20 |   |      | 0v  |      |      |
| 9  | 13 | MISO | IN  | 0 | 21 || 22 | 0 | IN  | GPIO. 6 | 6  | 25  |
| 11 | 14 | SCLK | IN  | 0 | 23 || 24 | 1 | IN  | CE0  | 10  | 8   |
|      |      | 0v  |      |   | 25 || 26 | 1 | IN  | CE1  | 11  | 7   |
| 0  | 30 | SDA.0 | IN  | 1 | 27 || 28 | 1 | IN  | SCL.0 | 31  | 1   |
| 5  | 21 | GPIO.21 | IN  | 1 | 29 || 30 |   |      | 0v  |      |      |
| 6  | 22 | GPIO.22 | IN  | 1 | 31 || 32 | 0 | IN  | GPIO.26 | 26 | 12  |
| 13 | 23 | GPIO.23 | IN  | 0 | 33 || 34 |   |      | 0v  |      |      |
| 19 | 24 | GPIO.24 | IN  | 0 | 35 || 36 | 1 | OUT | GPIO.27 | 27 | 16  |
| 26 | 25 | GPIO.25 | IN  | 0 | 37 || 38 | 0 | IN  | GPIO.28 | 28 | 20  |
|      |      | 0v  |      |   | 39 || 40 | 0 | IN  | GPIO.29 | 29 | 21  |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| BCM | wPi | Name | Mode | V | Physical | V | Mode | Name | wPi | BCM |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

La columna del centro “Physical” indica el orden físico de los pines. El verdaderamente importante es el número de la columna wPi (Wiringpi) que sera el numero con el que definiremos que pin queremos programar cuándo usamos los comandos GPIO.

Los pines GPIO tienen su propia librería de comandos que ya viene instalada con el sistema Raspbian. Estos comandos serán la parte más importante del script.

Código del script

```
#!/bin/bash

gpio mode 2 up
while true
do
    perm=$(grep "denegado" /home/pi/scripts_alibabox/clave)
    estado=$(gpio read 2)
    if [[ $estado == 1 && $perm == "denegado" ]]
    then
        ##Apertura sin permiso

        play /home/pi/scripts_alibabox/alarmanuclear.mp3 &
        nohup /home/pi/scripts_alibabox/pruebamail.sh
        until [[ $estado == 0 ]]
        do
            estado=$(gpio read 2)
            sleep 0
        done
    elif [[ $estado == 1 && $perm != "denegado" ]]
    then
        ##Permiso concedido

        until [[ $estado == 0 ]]
        do
            estado=$(gpio read 2)
            sleep 0
        done
    fi
done
```

Funcionamiento:

La primera línea activa el pin 2

Desde la segunda línea hasta la última se abre un bucle infinito.

La primera línea del bucle establece la variable “perm”(llave) que será el resultado del comando `grep “denegado” /home/pi/scripts_alibabox/clave` . Este comando comprueba si en el archivo indicado existe la palabra denegado. La segunda línea del bucle establece que el valor de la variable “estado” es el resultado del comando `gpio read 2` que muestra un valor dado por el pin 2 cuando está o no conectado.

Si el valor de la variable estado es 1 (puerta abierta) y la variable perm es igual a la palabra “denegado” reproduce la alarma y se ejecuta el script *foto.sh*

Después de completar la función de *foto.sh* el script entra en bucle, esperando a que la variable estado vuelva a ser 0 para continuar.

De esta manera evitas que se envíen mensajes infinitos hasta que se cierre la puerta (envía uno y se queda esperando)

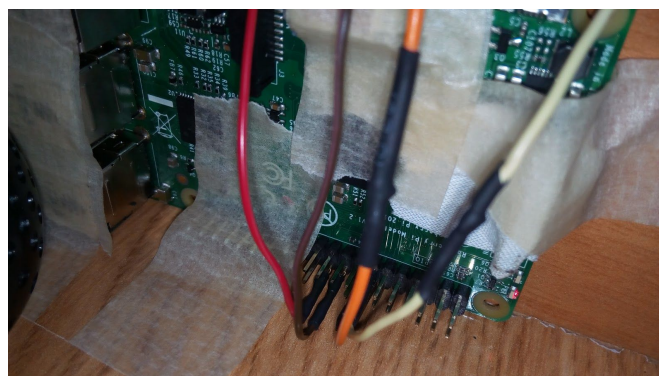
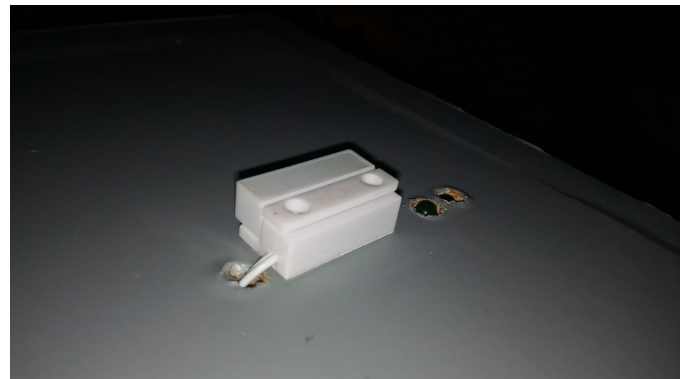
Después, la línea del “elif” establece que si la puerta se abre pero la llave está puesta, no hace nada y espera a que se vuelva a cerrar la puerta.

fin del bucle

Llave

Este script se encargará de establecer si hay permiso o no para abrir la caja utilizando una llave para definirlo.

A modo de llave utilizaremos otro “Reed Switch” como el que hemos usado para la puerta, pero este lo pondremos en la parte superior de la caja y lo conectaremos en los pines 18 y 20, siendo el pin 20 tierra.



El script escribirá, en función de si la llave está presente o no, la palabra “concedido” o “denegado” en un archivo que crearemos llamado “clave” en la misma carpeta donde tenemos el resto de scripts.

Empezaremos creando este archivo con el comando touch así:

```
touch /home/pi/scripts_alibabox/clave
```

Después, con el mismo comando crearemos el script y le daremos permisos de ejecución con chmod

```
touch /home/pi/scripts_alibabox/llave.sh  
chmod 777 /home/pi/scripts_alibabox/llave.sh
```

Hecho esto podemos proceder a introducir el código del script

Código del script

```
#!/bin/bash
```

```
gpio mode 5 up
```

```
while true
```

```
do
```

```
    estado=$(gpio read 5)
```

```
    if [[ $estado == 1 ]]
```

```
        then
```

```
            echo "denegado" > /home/pi/scripts_alibabox/clave
```

```
    elif [[ $estado == 0 ]]
```

```
        then
```

```
            echo "concedido" > /home/pi/scripts_alibabox/clave
```

```
    fi
```

```
done
```

La primera línea activa el pin 5, la segunda empieza un bucle infinito. Dentro del bucle primero se establece la variable estado, que es el resultado del comando *gpio read 5*. Si la

variable estado es igual a 1 (sin llave) escribirá “denegado” en el archivo indicado. Si la variable estado es igual a 0 (con llave) escribirá “concedido” en el archivo indicado.
fin del bucle

Archivo rc.local

Una vez hechos todos los scripts, para que nuestra Ali Babox funcione como debe tenemos que escribir en el archivo **rc.local** la orden de ejecutar los scripts necesarios.

rc.local es un script que se ejecuta al iniciar la máquina y al terminar su función se cierra. Como habréis podido pensar esto supone un problema ya que necesitamos que nuestro script lea el estado de la puerta de forma continua, así que tendremos que hacer que se ejecute en segundo plano, que no bloquee el cierre del script rc.local, y que nuestros scripts no se cierren al mismo tiempo que rc.local.

Primero nos aseguraremos de que el script tiene permisos de ejecución

```
chmod 777 /home/pi/scripts_alibabox/tuscript.sh
```

Recordamos que si teneis el script en un lugar diferente cambiad la ruta por la vuestra.

Después abrimos el archivo rc.local

```
sudo vi /etc/rc.local
```

Veremos que el propio archivo ya tiene una algo de código escrito, al final del código pero siempre y muy importante, **antes de la línea exit 0** escribimos la orden de ejecutar nuestro script. Debería quedar así:

```
#!/bin/sh
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# By default this script does nothing.

# Print the IP address
_IP=$(hostname -I) || true
if [ "$_IP" ]; then
printf "My IP address is %s\n" "$_IP"
```

fi

```
nohup /home/pi/scripts_alibabox/llave.sh &  
nohup /home/pi/scripts_alibabox/puerta.sh &  
play /home/pi/scripts_alibabox/alarmanuclear.mp3
```

exit 0

Estas líneas irán a buscar nuestros scripts y los ejecutarán.

El parámetro nohup sirve para que este script no se cierre al cerrarse el rc.local.

El parámetro & sirve para que el script se ejecute en segundo plano.

La penúltima línea “*play ...*” es para que la alarma suene al encenderse la máquina, a modo de aviso de que esta está lista.

Preparación caja Fuerte

Marquetería

la preparación de la caja fuerte que hemos llevado a cabo a sido la siguiente:

1- cogimos 2 trozos grandes de madera para cortar en 6 en las medidas de 44x25 Cm.



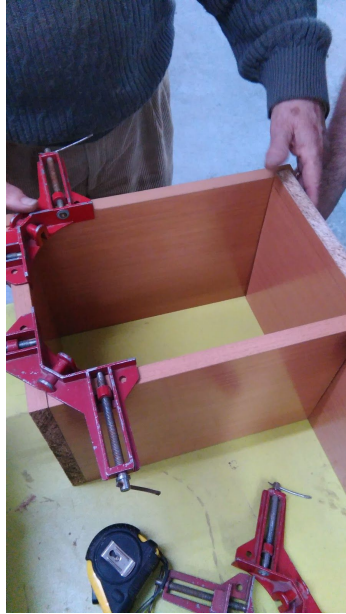
2- una vez cortada dicha madera, separamos los trozos con su pareja en total 6 trozos de los cuales 2 son laterales, 1 base, 1 tapa y acabamos con la puerta.



3- seguidamente, cogeremos los trozos de madera laterales con la tapa y base para iniciar así la estructura. primero cojeremos la madera lateral y la juntaremos con la madera lateral trasera y las juntaremos formando una L. incluso podemos coger 3 trozos y poner un sargento para que no se mueva la madera y poder pegar las 3 partes, nosotros el método que hemos cogido a sido el siguiente, cogimos 3 trozos laterales y con ayuda del abuelo de Manu juntamos las piezas con los sargentos.



4- una vez cogido las tres partes con el sargento, agregaremos la base de la caja más la parte lateral que nos falta para empezar ya a pegar los trozos.



5- una vez pegadas las cuatro partes que formarán el esqueleto nos faltaria poner la puerta que irá encajada en un lateral con una bisagra grande de esa forma conseguiremos que la caja pueda abrir y cerrar la puerta. finalmente así es como quedará la caja.



6- decidimos forrar la caja para darle una apariencia más seria y realista para que más o menos se parezca a una caja fuerte, el resultado fue bastante agradable incluso quedó mejor que no forrada.



7-lo que haremos a continuación será ir a la parte trasera de la caja y marcar con un bolígrafo o lápiz tres marcas, esas marcas serán los diferentes agujeros que tendremos que hacer para pasar el cable de internet, enchufe y para finalizar un USB.

8- una vez hechas las marcas de los agujeros, con ayuda de un taladro de mesa con mucho cuidado haremos los agujeros con una broca de 24 mm, la diferencia que tiene que haber entre agujero y agujero tiene que ser la suficiente para que los cables entre ellos no se molesten.



9- Cuando hayamos Creado los agujeros de la parte trasera de la caja fuerte lo miraremos desde adentro para ver como ha quedado,



como podemos observar en la siguiente fotografía no han quedado mal así que procederemos a poner los cables, antes de poner los cables tendréis que mirar si vuestra raspberry encaja bien en los agujeros en nuestro caso encaja a la perfección y la distribución de los agujeros será la siguiente, el primer agujero es por donde entrará el cable de electricidad para alimentar la raspberry, el segundo agujero un poco más ancho es para el cable de internet y para finalizar el último agujero es para USB.

10- ahora nos enfocaremos en la parte superior donde haremos dos agujeros pequeños con una broca de 15 mm para los leds, de esta manera los leds quedan fijos y no se moverán esta parte es opcional pero estéticamente queda bien, un poco más al lado haremos un agujero con una broca de 12mm para pasar los cables de la llave electrónica.

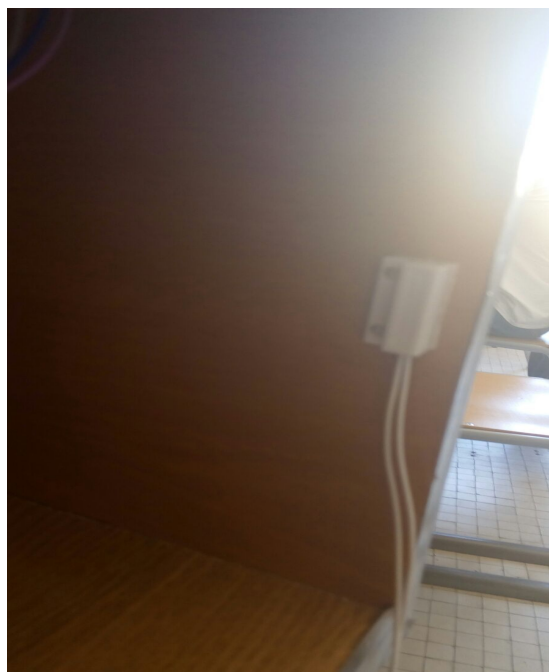
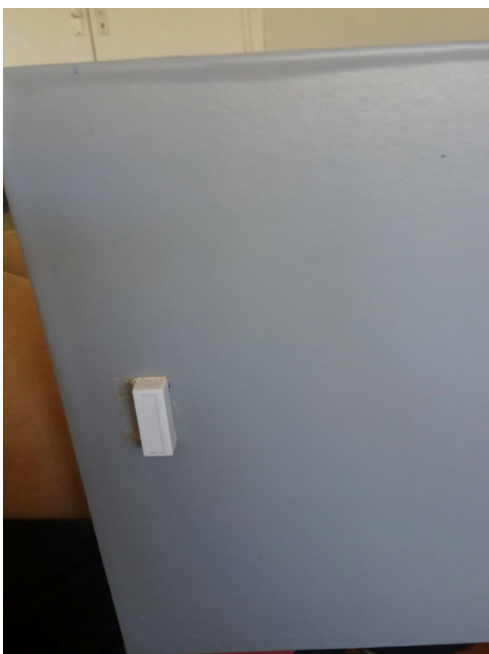


11- ya hemos acabado con la parte superior de la caja ahora nos enfocaremos en las entrañas de la caja. dentro de la caja no habrá que hacer ningún agujero sino tendremos que poner el said y el cierre magnético empezaremos por el said que la posición es diferente pero nosotros

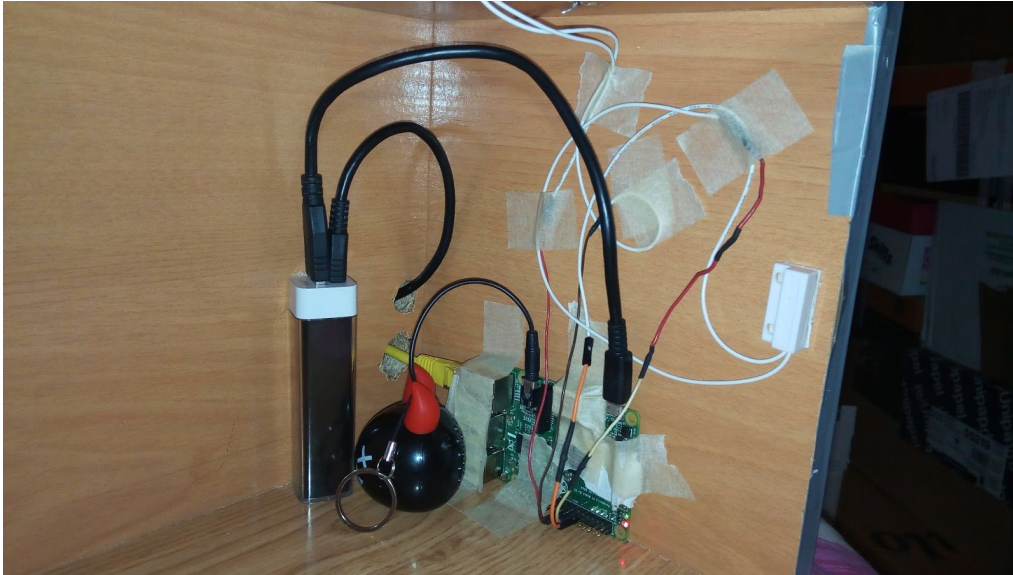
por comodidad lo hemos puesto en la parte inferior derecho al lado de los agujeros, ya que de esa manera el cable del said podrá llegar con facilidad a la raspberry



12- seguimos dentro de la caja ahora pondremos el cierre magnético uno en la puerta que será el que no va con electricidad, y en el lateral de adentro de la caja será el que va con electricidad, una vez hecho esto lo que hará el cierre magnético es lo siguiente si el cierre están los dos juntos enviara de respuesta un 0 eso quiere decir que está cerrado cuando el cierre se separe la información cambiaria y la respuesta será 1. de esta manera la raspberry sabrá cuando está la puerta abierta o no.



13- Cuando hayamos puesto el cierre magnético lo que tendremos que hacer es introducir la raspberry de manera que quede alineada con los agujeros de esta manera los cables quedarán bien a la hora de introducirlos.



14- por último y menos importante nos faltará introducir la cámara raspberry que estará colocada en el interior de la caja de esa manera dicha cámara podrá hacer una foto decente del presunto ladrón.