

Proyecto Final: Centro Multimedia

Daniela Ramírez Andrés

1. Introducción

Conocer el funcionamiento de un sistema embebido nos permite tener un mejor control sobre lo que deseamos realizar. Al ser de propósito específico, no existen complicaciones respecto a posibles fallos. La Raspberry Pi es una tarjeta microcontroladora que actúa como un sistema embebido. Para su funcionamiento, debe contar con un sistema operativo que nos permita controlar la Raspberry Pi. En este caso, usaremos Raspbian OS, que es una distribución basada en Debian. [1]

La Raspberry Pi nos permite crear, de manera efectiva y económica, un centro multimedia para reproducir contenido digital. Al contar con dispositivos periféricos de bajo costo, la Raspberry Pi permite la reproducción de estos medios digitales con un gestor gráfico simple, como VLC. Sin embargo, para contenido de streaming online, podemos acceder mediante un navegador web optimizado para la reproducción de video protegido con DRM. [2]

2. DRM

DRM (Gestión de Derechos Digitales) es un conjunto de tecnologías y herramientas utilizadas para controlar y gestionar el acceso y uso de contenido digital. Su propósito principal es proteger los derechos de los propietarios de medios, asegurando que el contenido se utilice de acuerdo con las licencias y restricciones establecidas. DRM previene la copia, distribución y modificación no autorizada de obras digitales, garantizando así que los creadores y distribuidores mantengan el control sobre cómo se utilizan sus productos.[4]

3. Chromium

Chromium es una base de código abierto para desarrollar un navegador web, mantenida por diversas compañías que posteriormente usan el código fuente para crear su propia versión de navegador con características adicionales. Este enfoque permite a los desarrolladores acceder a una base sólida y bien mantenida para construir sus propios navegadores web, adaptados a sus necesidades específicas o a las demandas del mercado.

4. Detección de inserción de dispositivos USB con pyudev

Según la documentación oficial, pyudev es una biblioteca desarrollada en Python puro y licenciada bajo LGPL, que actúa como un envoltorio para libudev, la biblioteca de gestión de dispositivos y hardware en sistemas Linux. Con pyudev, es posible enumerar dispositivos, consultar sus propiedades y atributos, y monitorizarlos de manera asíncrona. A continuación, se describen los pasos necesarios para detectar nuevos dispositivos:

- Importar las librerías necesarias.
- Definir una función `device_event` que maneje los eventos de dispositivos. Esta función se ejecuta cada vez que ocurre un evento de dispositivo y verifica si la acción es 'add' (agregar) y si el dispositivo pertenece al bus USB.
- Crear un contexto `pyudev`.
- Configurar un monitor para escuchar eventos de la subred 'usb'.
- Crear un observador (`MonitorObserver`) que ejecute la función `device_event` cada vez que ocurra un evento relevante.
- Iniciar el observador y entrar en un bucle infinito para mantener el script en ejecución.

5. Materiales

Para la realización de este proyecto, será necesario contar con una Raspberry Pi 3B o superior con el sistema operativo Raspbian OS Lite, basado en Debian. Además, se requerirá un monitor, una microSD, un dispositivo apuntador para controlar el centro multimedia y una unidad USB que contenga archivos de reproducción digital, como imágenes, videos y música.

Importante

Es importante destacar que el sistema operativo debe estar configurado sin un entorno de escritorio, es decir, sin un gestor gráfico.

6. Instrucciones

1. Descarga y carga de sistema operativo
2. Configuración para servicios en línea
3. Configuración para USB

6.1. Descarga y carga de sistema operativo

6.1.1. Descarga de RaspbianOS Lite

Deberá ingresar a la página web <https://www.raspberrypi.com/software/operating-systems/>, donde podrá descargar una versión de RaspbianOS Lite. Si posee una Raspberry Pi 3B o superior, se recomienda utilizar la versión de 64 bits. Es importante tener en cuenta que, mientras más reciente sea la versión, es más probable que contenga algunos errores que aún no han sido probados. Por esta razón, para nuestro caso, se recomienda utilizar una versión más antigua. En este caso, contamos con la versión de febrero de 2020.

6.1.2. Escribir imagen iso en la microSD

Una vez descargado el sistema operativo contara con una carpeta comprimida debera descomprimirla y tendra un archivo tipo iso. Para escribir la imagen en la microSD se requiere de un programa externo que nos ayude dependiendo del sistema operativo con el que contemos

- **Escribiendo iso en Linux**

1. Para escribir una imagen ISO, es necesario contar con el programa Etcher. Para descargar e instalarlo, ejecutaremos las siguientes líneas:

```
$ wget https://github.com/balena-io/etcher/releases/download/v1.5.126/balena-etcher-electron-1.5.126-linux-x64.zip
$ unzip balena-etcher-electron-1.5.126-linux-x64.zip
# mv balenaEtcher-electron-1.5.126-x86_64.AppImage /usr/local/bin/balenaEtcher
# chmod +x /usr/local/bin/balenaEtcher
```

2. En la interfaz de Etcher, selecciona la imagen de disco que desees grabar en tu tarjeta SD.
3. Selecciona el dispositivo USB o tarjeta SD en el que desees grabar la imagen.

Advertencia

Ten en cuenta que esto borrará todos los datos en el dispositivo seleccionado, así que asegúrate de elegir el dispositivo correcto.

4. Haz clic en el botón "Flash!" para iniciar el proceso de grabación. Es posible que necesites ingresar tu contraseña de superusuario (root) para permitir que Etcher acceda al dispositivo seleccionado.
5. Una vez que Etcher haya terminado de grabar la imagen en el dispositivo, verás un mensaje de confirmación. Puedes cerrar Etcher y retirar el dispositivo USB o tarjeta SD de forma segura.

- **Escribiendo iso en Windows**

1. Entra a la página <https://sourceforge.net/projects/win32diskimager/> donde encontrarás la descarga de Win32 Disk Imager, el cual nos servirá para grabar nuestra imagen.
2. Una vez que se haya completado la descarga, haz doble clic en el archivo .exe descargado para iniciar el proceso de instalación.
3. Una vez instalado, debemos abrirlo. Haz clic en el icono de la carpeta junto al campo "Imagen de disco" para seleccionar la imagen de disco que desees escribir en tu dispositivo de almacenamiento.
4. Haz clic en la lista desplegable de "Dispositivo" para seleccionar el dispositivo de almacenamiento al que desees escribir la imagen de disco.

5. Una vez que hayas seleccionado la imagen de disco y el dispositivo de almacenamiento, haz clic en el botón `.Escribir` para comenzar a escribir la imagen en el dispositivo.
6. Ahora puedes cerrar Win32 Disk Imager y desconectar tu dispositivo de almacenamiento y conectarlo en la Raspberry pi.

6.1.3. Configuraciones de RaspbianOS Lite

Una vez conectada la microSD a nuestra Raspberry Pi, debemos proceder a conectarla a la corriente usando un cable tipo C. Además, necesitaremos un adaptador micro HDMI a VGA y un cable VGA para conectar nuestro monitor. También haremos uso de un teclado.

Una vez que todo esté conectado, en nuestro monitor aparecerá un menú de configuración donde elegiremos el tipo de idioma para nuestro teclado. También se nos pedirá que creamos un usuario y una contraseña. Al ser un sistema operativo sin interfaz gráfica, lo único que veremos será una terminal donde trabajaremos.

a. Configuración de red alámbrica `eth0` con IP estática

Como mencionamos anteriormente, este proyecto está realizado en una versión más antigua de Raspbian OS, por lo cual configuraremos nuestra red cableada modificando el archivo `dhcpcd.conf`.

Tome en cuenta que los comandos marcados con `#` deben ejecutarse con permisos de super-usuario (`sudo`).

1. Abra el archivo `dhcpcd.conf` que se encuentra en `/etc/dhcpcd.conf`.

```
# sudo nano /etc/dhcpcd.conf
```

2. Ahora debemos editar el archivo para configurar nuestra IP local. Localice la línea `# Example static IP configuration:`. Verá un código similar al siguiente:

```
# Example static IP configuration:
#interface eth0
#static ip_address=192.168.0.10/24
#static ip6_address=fd51:42f8:caae:d92e::ff/64
#static routers=192.168.0.1
#static domain_name_servers=192.168.0.1 8.8.8.8 fd51:42f8:caae:d92e::1
```

3. Edite el archivo con su IP local como se muestra a continuación:

```
# Example static IP configuration:
interface eth0
static ip_address=192.168.100.1/24
static routers=192.168.100.1
static domain_name_servers=132.248.204.1 132.248.10.2 8.8.8.8
```

donde

- `static_ip_address` es la dirección IP de la Raspberry Pi, seguida por el código de la máscara de subred (por ejemplo, 24 equivale a 255.255.255.0).
- `static_routers` es la dirección IP del enrutador de la red local, que actúa como la puerta de enlace hacia Internet.
- `static_domain_name_servers` es la dirección de los servidores de nombres de dominio (DNS), utilizados para la resolución de nombres en Internet.

6.2. Configuración para servicios en línea

Para los servicios en línea debemos contar con un navegador y tener correctamente configurado el DRM el cual nos permitira acceder a este contenido.

6.2.1. Instalación de chromium

- Para poder ejecutar Chromium, necesitaremos una interfaz gráfica sencilla que nos permita visualizar el navegador. Para esto, utilizaremos `xinit` o `startx`. [5]

```
# apt-get update
# apt-get install xinit
```

- Una vez instalado, procederemos a instalar Chromium ejecutando lo siguiente:

```
# apt-get install chromium-browser
```

- Acceder al navegador

Vamos a verificar que tenemos acceso al navegador con la siguiente línea:

```
$ startx /bin/chromium-browser
```

6.2.2. Configuración de DRM

Para poder visualizar el contenido de estas plataformas, es necesario configurarlas para que el DRM sea accesible. Esto implica establecer la configuración adecuada y, en algunos casos, instalar y configurar software adicional, como navegadores web o reproductores multimedia compatibles con DRM. [3]

- Actualización de paquetes

```
# apt update && sudo apt full-upgrade -y
```

- instalar las librerías widevine

```
# apt upgrade && sudo apt install libwidevinecdm0 -y
```

6.3. Configuración para USB

7. Resultados

8. Conclusiones

9. Cuestionario

Referencias

- [1] Alejandro Calderon Mateos. Tutoria del trabajo de fin de grado "nube multimedia personal: un caso de estudio con raspberry pi, owncloud y xbmc" de sergio fernandez romero. En Linea, 2014.
- [2] C. González. Aplicaciones orientadas a la domótica con raspberry pi. [En Linea], 2015. Recuperado el 19 de mayo de 2024.
- [3] Sergio. Ver netflix en raspberry pi es ahora más fácil que nunca gracias a los repositorios de raspberry pi os. En Linea, 2021. Publicado el 1 de abril de 2021.
- [4] Subodh. Significado de la protección de contenido drm y cómo proteger un archivo de vídeo con drm. En Linea, 2023. Recuperado el 19 de mayo de 2024.
- [5] Usuario. Raspberry pi os lite con startx. En Linea, 2022. Mensaje en línea en el foro de Raspberry Pi. Publicado el 18 de diciembre de 2022 a las 3:30 a.m.

10. Apéndice