



# IT Department Technical Handbook

**PREPARED BY:**

GROUP 1

# Table of Contents

- 1 Introduction
- 2 Purpose of the Manual
- 3 Construction Worker Functions
- 3 Job Guidelines
- 3 General Responsibilities
- 4 Company Safety Program
- 5 Company Health and Environment Programs
- 6 Company Cleaning Programs
- 7 Construction Worker Employee Orientation
- 8 Incident Investigating and Reporting
- 9 Auditing and Inspecting

## **1. Lanzar la instancia**

S.O.

**Inicio rápido**

Amazon Linux  
aws

macOS  
Mac

**Ubuntu**  
ubuntu

Windows  
Microsoft

Red Hat  
Red Hat

SUSE Linux  
SUSE Linux

Debian  
debian

Incluir de AMI de AWS, Marketplace y la comunidad

**Imágenes de máquina de Amazon (AMI)**

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type  
 ami-0f9de6e2d2f067fca (64 bits (x86)) / ami-0967e5535761d839e (64 bits (Arm))  
 Virtualización: hvm    Activado para ENA: true    Tipo de dispositivo raíz: ebs

Apto para la capa gratuita

## Ubuntu Server 22.04 LTS

**▼ Tipo de instancia** [Información](#) | [Obtener asesoramiento](#)

**Tipo de instancia**

t2.large  
 Familia: t2    2 vCPU    8 GiB Memoria    Generación actual: true  
 Bajo demanda Windows base precios: 0.1208 USD por hora    Bajo demanda RHEL base precios: 0.1216 USD por hora  
 Bajo demanda SUSE base precios: 0.1928 USD por hora    Bajo demanda Ubuntu Pro base precios: 0.0963 USD por hora  
 Bajo demanda Linux base precios: 0.0928 USD por hora

☐ Todas las generaciones  
[Comparar tipos de instancias](#)

[Se aplican costos adicionales a las AMI con software preinstalado](#)

## Instancia t2.large

**▼ Par de claves (inicio de sesión)** [Información](#)

Puede utilizar un par de claves para conectarse de forma segura a la instancia. Asegúrese de que tiene acceso al par de claves seleccionado antes de lanzar la instancia.

**Nombre del par de claves - obligatorio**

## Par de claves

**▼ Configuraciones de red** [Información](#) [Editar](#)

**Red** [Información](#)

vpc-062876eac3f4fa4c1

**Subred** [Información](#)

Sin preferencias (subred predeterminada en cualquier zona de disponibilidad)

**Asignar automáticamente la IP pública** [Información](#)

☐ Habilitar

[Se aplican cargos adicionales cuando no se cumplen los límites del nivel gratuito](#)

**Firewall (grupos de seguridad)** [Información](#)

Un grupo de seguridad es un conjunto de reglas de firewall que controlan el tráfico de la instancia. Agregue reglas para permitir que un tráfico específico llegue a la instancia.

☐ Crear grupo de seguridad
 ☒ Seleccionar un grupo de seguridad existente

**Grupos de seguridad comunes** [Información](#)

default sg-04dbb93200f11b608 ✕

VPC: vpc-062876eac3f4fa4c1

Los grupos de seguridad que agrega o elimine aquí se agregarán a todas las interfaces de red o se eliminarán de ellas.

## Modo por defecto en el grupo de seguridad

▼ **Configurar almacenamiento** Información Avanzado

1x  GiB  Volumen raíz, No cifrado

❗ Los clientes que cumplan los requisitos de la capa gratuita pueden obtener hasta 30 GB de almacenamiento magnético o de uso general (SSD) de EBS

[Agregar un nuevo volumen](#)

The selected AMI contains Instance store volumes, however the Instance does not allow any Instance store volumes. None of the Instance store volumes from the AMI will be accessible from the Instance

ⓘ Haga clic en actualizar para ver la Información de la copia de seguridad  
Las etiquetas que asigne determinan si alguna política de Data Lifecycle Manager realizará una copia de seguridad de la instancia.

0 x sistemas de archivos Editar

## Almacenamiento de la máquina

**Editar reglas de entrada** Información

Las reglas de entrada controlan el tráfico entrante que puede llegar a la instancia.

Reglas de entrada	ID de la regla del grupo de seguridad	Tipo	Protocolo	Intervalo de puertos	Origen	Descripción: opcional	
	sgr-02b70a44ed15109ee	TCP personalizado	TCP	8000	Anywh...	Icecast HTTP streaming publico	<a href="#">Eliminar</a>
	sgr-02a4ee29cb397eadd	SSH	TCP	22	Anywh...	Conexion remota SSH	<a href="#">Eliminar</a>
	sgr-047a5e7f00f2d678b	RDP	TCP	3389	Anywh...	Conexion remota RDP	<a href="#">Eliminar</a>
	-	UDP personalizado	UDP	5004	Anywh...	RTP Streaming standard	<a href="#">Eliminar</a>

[Agregar regla](#)

## Reglas de entrada

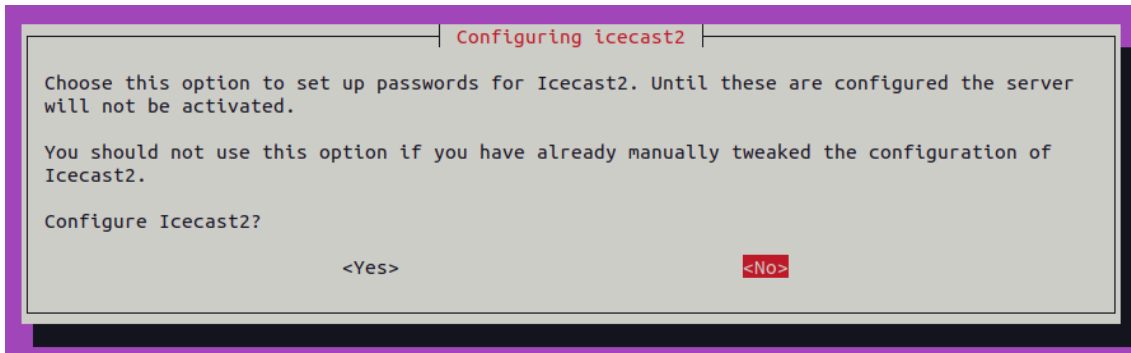
## 2. Configuración servicio de audio

```
ubuntu@ip-172-31-38-203:~$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [127 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 c-n-f Metadata [286 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [217 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse Translation-en [112 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 c-n-f Metadata [8372 B]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [2591 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [419 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [18.5 kB]
B]
```

## Update

```
ubuntu@ip-172-31-38-203:~$ sudo apt install -y icecast2 darkice vnstat iftop iperf3 ufw
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
icecast2 is already the newest version (2.4.4-4build1).
iftop is already the newest version (1.0~pre4-7).
vnstat is already the newest version (2.9-1).
darkice is already the newest version (1.3-0.3).
ufw is already the newest version (0.36.1-4ubuntu0.1).
iperf3 is already the newest version (3.9-1+deb11u1build0.22.04.1).
0 upgraded, 0 newly installed, 0 to remove and 68 not upgraded.
ubuntu@ip-172-31-38-203:~$
```

Instalamos el Icecast2 y otras utilidades (darkice, vnstat, iftop, iperf3 y ufw)



En la instalación de Icecast2 seleccionamos no poner contraseñas para configurarlo manualmente

```
ubuntu@ip-172-31-38-203:~$ sudo sed -i 's/ENABLE=false/ENABLE=true/' /etc/default/icecast2
ubuntu@ip-172-31-38-203:~$
```

Activamos icecast2

```
GNU nano 6.2 /etc/icecast2/icecast.xml
<icecast>
  <location>Server</location>
  <admin>admin@localhost.com</admin>

  <limits>
    <clients>100</clients>
    <sources>2</sources>
    <threadpool>5</threadpool>
    <queue-size>524288</queue-size>
    <client-timeout>30</client-timeout>
    <header-timeout>15</header-timeout>
    <source-timeout>10</source-timeout>
  </limits>

  <authentication>
    <source-password>pirineus</source-password>
    <admin-user>admin</admin-user>
    <admin-password>pirineus</admin-password>
  </authentication>

  <hostname>localhost</hostname>

  <listen-socket>
    <port>8000</port>
  </listen-socket>

  <mount>
    <mount-name>/live.mp3</mount-name>
  </mount>

  <paths>
    <basedir>/usr/share/icecast2</basedir>
    <logdir>/var/log/icecast2</logdir>
    <webroot>/usr/share/icecast2/web</webroot>
    <adminroot>/usr/share/icecast2/admin</adminroot>
    <alias source="/" destination="/status.xml"/>
  </paths>

  <logging>
    <accesslog>access.log</accesslog>
    <errorlog>error.log</errorlog>
    <loglevel>3</loglevel>
    <logsize>10000</logsize>
  </logging>

  <security>
    <chroot>0</chroot>
  </security>

  <http-headers>
    <header name="Access-Control-Allow-Origin" value="*" />
  </http-headers>
</icecast>
```

Configuración del fichero /etc/icecast2/icecast.xml

```
GNU nano 6.2 /etc/darkice.cfg
[general]
duration          = 0
bufferSecs        = 5
reconnect         = yes

[input]
device            = default
sampleRate        = 44100
bitsPerSample     = 16
channel           = 2

[icecast2-0]
bitrateMode       = cbr
format            = mp3
bitrate           = 128
server            = localhost
port              = 8000
password          = pirineus
mountPoint        = live.mp3
name              = Stream d'àudio
```

Configuración del fichero /etc/darkice.cfg

```
ubuntu@ip-172-31-35-226:~$ sudo ufw allow 8000/tcp
Rules updated
Rules updated (v6)
ubuntu@ip-172-31-35-226:~$ sudo ufw allow 22/tcp
Rules updated
Rules updated (v6)
ubuntu@ip-172-31-35-226:~$
```

```
ubuntu@ip-172-31-35-226:~$ sudo ufw status
Status: active

To Action From
--
8000/tcp ALLOW Anywhere
22/tcp ALLOW Anywhere
8000/tcp (v6) ALLOW Anywhere (v6)
22/tcp (v6) ALLOW Anywhere (v6)

ubuntu@ip-172-31-35-226:~$
```

Abrimos los puertos 8000 y 22

```
ubuntu@ip-172-31-35-226:~$ sudo systemctl restart icecast2.service
ubuntu@ip-172-31-35-226:~$ sudo systemctl enable icecast2.service
icecast2.service is not a native service, redirecting to systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable icecast2
ubuntu@ip-172-31-35-226:~$ sudo systemctl status icecast2.service
● icecast2.service - LSB: Icecast2 streaming media server
   Loaded: loaded (/etc/init.d/icecast2; generated)
   Active: active (running) since Wed 2025-05-21 06:29:11 UTC; 11s ago
     Docs: man:systemd-sysv-generator(8)
    Tasks: 4 (limit: 9505)
   Memory: 3.1M
      CPU: 27ms
   CGroup: /system.slice/icecast2.service
           └─1522 /usr/bin/icecast2 -b -c /etc/icecast2/icecast.xml

May 21 06:29:11 ip-172-31-35-226 systemd[1]: Starting LSB: Icecast2 streaming media server...
May 21 06:29:11 ip-172-31-35-226 icecast2[1516]: * Starting streaming media server icecast2
May 21 06:29:11 ip-172-31-35-226 icecast2[1522]: [2025-05-21 06:29:11] WARN CONFIG/_parse_root Warn
May 21 06:29:11 ip-172-31-35-226 systemd[1]: Started LSB: Icecast2 streaming media server.
May 21 06:29:11 ip-172-31-35-226 icecast2[1516]: ...done.
lines 1-15/15 (END)
```

Reiniciamos, restablecemos y revisamos el estado del servicio

## 2.1 Configuración de la tarjeta de audio

Para poder iniciar correctamente el darkice necesitamos configurar una tarjeta de audio para que no de problemas, para ello, primero tenemos que instalar "libmp3lame-dev"

```
ubuntu@ip-172-31-47-116:~$ sudo apt-get install libmp3lame-dev
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
libmp3lame-dev is already the newest version (3.100-3build2).
0 upgraded, 0 newly installed, 0 to remove and 68 not upgraded.
ubuntu@ip-172-31-47-116:~$
```

Después de esto, tenemos que hacer un reboot de la máquina y al iniciar ejecutar el comando `uname -r` para verificar que el kernel que se utiliza es el generic que hemos generado nosotros

```
ubuntu@ip-172-31-47-116:~$ uname -r
5.15.0-140-generic
ubuntu@ip-172-31-47-116:~$
```

En caso de que no aparezca el que hemos generado, nosotros tendremos que ir a la ubicación `/lib/modules` para verificar que se haya generado correctamente

```
ubuntu@ip-172-31-47-116:/lib/modules$ ls
5.15.0-140-generic 6.8.0-1024-aws
ubuntu@ip-172-31-47-116:/lib/modules$
```



Si aparece en la ubicación, tendremos que ir al fichero de grub /etc/default/grub y en la línea GRUB\_DEFAULT tendremos que cambiar el 0 por la siguiente línea:

"Advanced options for Ubuntu>Ubuntu, with Linux 5.15.0-140-generic"

Poniendo el nombre del generic que tengamos, en nuestro caso, el 5.15.0-140-generic

```
GNU nano 6.2 /etc/default/grub
# If you change this file, run 'update-grub' afterwards to update
# /boot/grub/grub.cfg.
# For full documentation of the options in this file, see:
# info -f grub -n 'Simple configuration'

GRUB_DEFAULT="Advanced options for Ubuntu>Ubuntu, with Linux 5.15.0-140-generic"
GRUB_TIMEOUT_STYLE=hidden
GRUB_TIMEOUT=0
GRUB_DISTRIBUTOR=`lsb_release -i -s 2> /dev/null || echo Debian`
GRUB_CMDLINE_LINUX_DEFAULT="quiet splash"
GRUB_CMDLINE_LINUX=""
```

Una vez modificado el fichero y guardado, tenemos que hacer un update del grub con el comando sudo update-grub

Después volvemos a hacer un reboot de la máquina, y al iniciar ejecutamos el comando sudo modprobe snd-aloop

```
ubuntu@ip-172-31-47-116:/lib/modules$ sudo modprobe snd-aloop
ubuntu@ip-172-31-47-116:/lib/modules$
```

En caso de que no nos muestre salida es que está correcto, comprobaremos igualmente con el comando lsmod | grep snd\_aloop que se haya ejecutado correctamente

```
ubuntu@ip-172-31-47-116:/lib/modules$ lsmod | grep snd_aloop
snd_aloop                36864  0
snd_pcm                  143360  1 snd_aloop
snd_timer                 40960  2 snd_aloop,snd_pcm
snd                      106496  3 snd_timer,snd_aloop,snd_pcm
ubuntu@ip-172-31-47-116:/lib/modules$
```

En caso de ser así podemos continuar, si el comando sudo modprobe snd-aloop no se ha ejecutado correctamente o ha dado algún fallo, habría que volver a repetir el proceso.

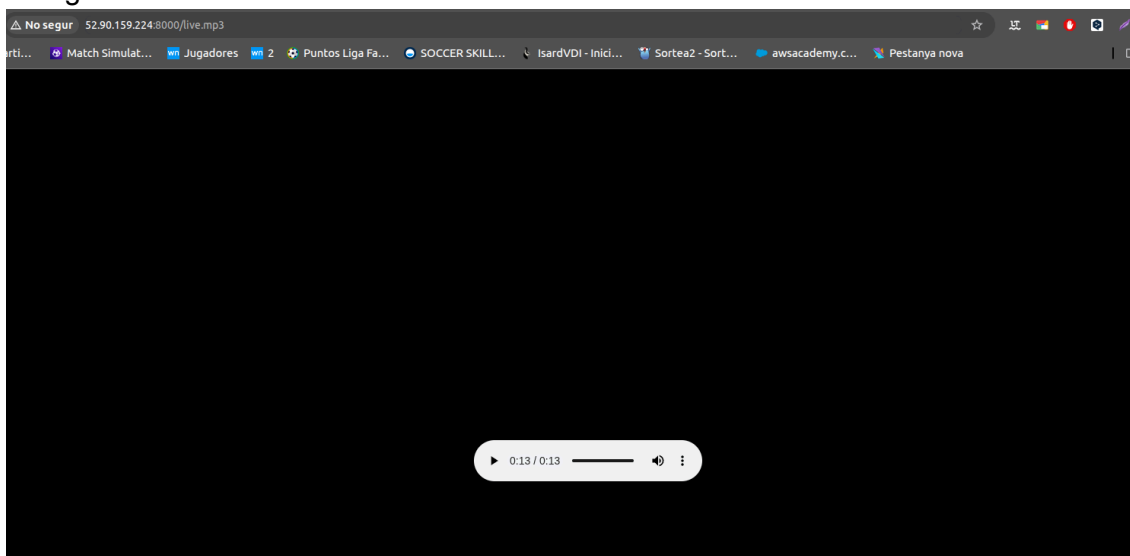
Una vez revisado el anterior paso, podemos ejecutar el comando sudo darkice, y nos deberá de mostrar lo siguiente

```
ubuntu@ip-172-31-47-116:/lib/modules$ sudo darkice
DarkIce 1.3 live audio streamer, http://code.google.com/p/darkice/
Copyright (c) 2000-2007, Tyrell Hungary, http://tyrell.hu/
Copyright (c) 2008-2013, Akos Maroy and Rafael Diniz
This is free software, and you are welcome to redistribute it
under the terms of The GNU General Public License version 3 or
any later version.

Using config file: /etc/darkice.cfg
Using ALSA DSP input device: default
Using POSIX real-time scheduling, priority 4

```

En este caso está correctamente configurado, ahora simplemente, para poder acceder al servicio mediante HTTP, deberemos de poner en nuestro navegador `http://<tu-IP>:8000/live.mp3`, y nos debería de mostrar el fichero de audio que hayamos configurado en la ruta



Y con esto estaría configurado el servicio de audio

## 2.2 Como generar grabaciones de audio

Para grabar audio, tendremos que ejecutar el siguiente comando

`$sudo arecord -D default -f cd <nombre del fichero>`

```
ubuntu@ip-172-31-47-116:/usr/share/icecast2/web$ sudo arecord -D default -f cd live.mp3
Recording WAVE 'live.mp3' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
█
```

La ruta en la que hay que tener los archivos de audio es la siguiente `/etc/icecast2/web`, pero hay que enlazar los archivos a `/usr/share/icecast2/web` para que funcione

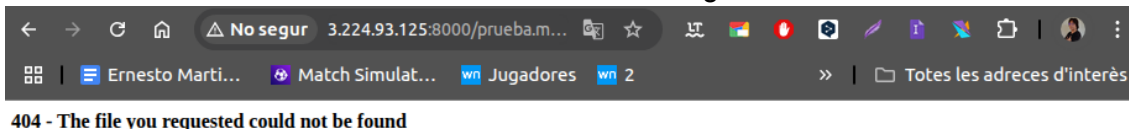
```
ubuntu@ip-172-31-47-116:/usr/share/icecast2/web$
/usr/share/icecast2/web
ubuntu@ip-172-31-47-116:/usr/share/icecast2/web$

ubuntu@ip-172-31-47-116:/usr/share/icecast2/web$ ls -la
total 23436
drwxr-xr-x 2 root root 4096 May 23 06:38 .
drwxr-xr-x 4 root root 4096 May 21 08:18 ..
lrwxrwxrwx 1 root root 26 Dec 8 2021 auth.xml -> /etc/icecast2/web/auth.xml
lrwxrwxrwx 1 root root 30 May 23 06:38 drakukeo.mp3 -> /etc/icecast2/web/drakukeo.mp3
-rw-r--r-- 1 root root 3959 Dec 8 2021 icecast.png
-rw-r--r-- 1 root root 1876 Dec 8 2021 key.png
-rw-r--r-- 1 root root 22349484 May 23 06:34 live.mp3
lrwxrwxrwx 1 root root 36 Dec 8 2021 server_version.xml -> /etc/icecast2/web/server_version.xml
sl
lrwxrwxrwx 1 root root 33 Dec 8 2021 status-json.xml -> /etc/icecast2/web/status-json.xml
lrwxrwxrwx 1 root root 28 Dec 8 2021 status.xml -> /etc/icecast2/web/status.xml
lrwxrwxrwx 1 root root 27 Dec 8 2021 style.css -> /etc/icecast2/web/style.css
-rw-r--r-- 1 root root 1613084 May 21 10:00 test.wav
-rw-r--r-- 1 root root 1671 Dec 8 2021 tunein.png
-rw-r--r-- 1 root root 8785 Dec 8 2021 xml2json.xslt
ubuntu@ip-172-31-47-116:/usr/share/icecast2/web$ █
```

Podemos enlazar los archivos con el siguiente comando

```
ubuntu@ip-172-31-47-116:/etc/icecast2/web$ sudo ln -s /etc/icecast2/web/drakukeo.mp3 /usr/share/icecast2/web/drakukeo.mp3
ubuntu@ip-172-31-47-116:/etc/icecast2/web$ █
```

En caso de no estar enlazado en la ruta nos dará el siguiente error



## 2.3 Configuración del script

Para evitar tener que configurar la tarjeta de audio siempre que entramos para que funcione darkice, podemos crear un script sencillo con los comandos necesarios para iniciar darkice

```
GNU nano 6.2                                iniciar_darkice.sh
#!/bin/bash
#Script para configurar la tarjeta de audio

sudo modprobe snd-aloop
sudo darkice
```

Y para grabar audio y evitar tener que crear nosotros mismos los enlaces, podemos generar también el siguiente script

```
GNU nano 6.2                                grabar_audio_darkice.sh
#!/bin/bash
#Script para grabar con darkice

# Solicitar el nombre del archivo
read -p "Introduce el nombre del archivo de audio (sin extensión): " nombre

# Agregar extensión .mp3 si no está incluida
archivo="$nombre.mp3"

# Ruta de destino
ruta_origen="/etc/icecast2/web/$archivo"
ruta_enlace="/usr/share/icecast2/web/$archivo"

# Grabar audio
echo "Grabando audio. Pulsa Ctrl+C para detener la grabación..."
sudo arecord -D default -f cd "$ruta_origen"

# Verificar si la grabación fue exitosa
if [ -f "$ruta_origen" ]; then
    echo "Grabación completada. Creando enlace simbólico..."
    sudo ln -sf "$ruta_origen" "$ruta_enlace"
    echo "Enlace creado: $ruta_enlace -> $ruta_origen"
else
    echo "Error: no se encontró el archivo grabado en $ruta_origen"
fi
```

Lo que hará, básicamente, es solicitarnos el nombre del fichero sin la extensión, para luego añadirla, comenzara a grabar el audio, indicando que para finalizar la grabación usemos Ctrl+C, y luego confirmara si se ha podido generar y enlazar correctamente el fichero

### 3. Servidor de vídeo

Primero que todo haremos un update

```
ubuntu@ip-172-31-47-116:~$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [127 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [2591 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1208 kB]
Get:7 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [2346 kB]
Get:8 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [974 kB]
Fetched 7502 kB in 1s (5486 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
26 packages can be upgraded. Run 'apt list --upgradable' to see them.
ubuntu@ip-172-31-47-116:~$
```

Luego instalaremos el siguiente paquete

`sudo apt install ffmpeg -y`

```
ubuntu@ip-172-31-47-116:/etc/icecast2/web$ sudo apt install ffmpeg -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ffmpeg is already the newest version (7:4.4.2-0ubuntu0.22.04.1).
0 upgraded, 0 newly installed, 0 to remove and 25 not upgraded.
```

Abriremos los siguientes puertos

`sudo ufw allow 5000/udp`

`sudo ufw allow 5004/udp`

```
ubuntu@ip-172-31-47-116:~$ sudo ufw allow 5000/udp
sudo ufw allow 5004/udp
Rule added
Rule added (v6)
Rule added
Rule added (v6)
ubuntu@ip-172-31-47-116:~$
```

Comprobamos que se hayan cambiado las reglas correctamente

```
ubuntu@ip-172-31-47-116:~$ sudo ufw status
Status: active

To Action From
--
8000/tcp ALLOW Anywhere
22/tcp ALLOW Anywhere
5000/udp ALLOW Anywhere
5004/udp ALLOW Anywhere
8000/tcp (v6) ALLOW Anywhere (v6)
22/tcp (v6) ALLOW Anywhere (v6)
5000/udp (v6) ALLOW Anywhere (v6)
5004/udp (v6) ALLOW Anywhere (v6)

ubuntu@ip-172-31-47-116:~$
```

#### 3.1 Emitir un video

Para poder emitir un video tendremos que tener primero el video descargado en nuestro sistema, para ello usaremos el comando `wget` para descargarnos el video una vez descargado, simplemente tendremos que ejecutar el siguiente comando

```
ffmpeg -re -i ruta/video -c:v libvpx -b:v 800k -c:a libvorbis -b:a 128k -f webm
-content_type video/webm
icecast://source:pirineus@3.224.93.125:8000/stream.webm
```

```
ubuntu@ip-172-31-47-116:/etc/icecast2/web$ ffmpeg -re -i video/rickroll.mp4 -c:v libvpx -b:v 800k
-c:a libvorbis -b:a 128k -f webm -content_type video/webm icecast://source:pirineus@3.224.93.125
:8000/stream.webm

libavutil      56. 70.100 / 56. 70.100
libavcodec     58.134.100 / 58.134.100
libavformat    58. 76.100 / 58. 76.100
libavdevice    58. 13.100 / 58. 13.100
libavfilter     7.110.100 / 7.110.100
libswscale     5.  9.100 / 5.  9.100
libswresample  3.  9.100 / 3.  9.100
libpostproc   55.  9.100 / 55.  9.100
Input #0, mov,mp4,m4a,3gp,3g2,mj2, from 'video/rickroll.mp4':
Metadata:
  major_brand      : mp42
  minor_version    : 0
  compatible_brands: isommp42
  creation_time    : 2017-01-31T19:48:03.000000Z
  Duration: 00:00:53.92, start: 0.000000, bitrate: 405 kb/s
Stream #0:0(und): Video: h264 (Main) (avc1 / 0x31637661), yuv420p, 640x360 [SAR 1:1 DAR 16:9], 276
kb/s, 30 fps, 30 tbr, 90k tbn, 60 tbc (default)
Metadata:
  creation_time    : 2017-01-31T19:48:03.000000Z
  handler_name     : ISO Media file produced by Google Inc. Created on: 01/31/2017.
  vendor_id       : [0][0][0][0]
Stream #0:1(und): Audio: aac (LC) (mp4a / 0x6134706D), 44100 Hz, stereo, fltp, 125 kb/s (default)
Metadata:
  creation_time    : 2017-01-31T19:48:03.000000Z
  handler_name     : ISO Media file produced by Google Inc. Created on: 01/31/2017.
  vendor_id       : [0][0][0][0]
Stream mapping:
  Stream #0:0 -> #0:0 (h264 (native) -> vp8 (libvpx))
  Stream #0:1 -> #0:1 (aac (native) -> vorbis (libvorbis))
Press [q] to stop, [?] for help
[libvpx @ 0x55eda5f02940] v1.11.0
Output #0, webm, to 'icecast://source:pirineus@3.224.93.125:8000/stream.webm':
Metadata:
  major_brand      : mp42
  minor_version    : 0
  compatible_brands: isommp42
  encoder         : Lavf58.76.100
Stream #0:0(und): Video: vp8, yuv420p(progressive), 640x360 [SAR 1:1 DAR 16:9], q=2-31, 800 kb/s, 3
0 fps, 1k tbn (default)
Metadata:
  creation_time    : 2017-01-31T19:48:03.000000Z
  handler_name     : ISO Media file produced by Google Inc. Created on: 01/31/2017.
  vendor_id       : [0][0][0][0]
  encoder         : Lavc58.134.100 libvpx
Side data:
  cpb: bitrate max/min/avg: 0/0/0 buffer size: 0 vbv_delay: N/A
Stream #0:1(und): Audio: vorbis, 44100 Hz, stereo, fltp, 128 kb/s (default)
Metadata:
  creation_time    : 2017-01-31T19:48:03.000000Z
  handler_name     : ISO Media file produced by Google Inc. Created on: 01/31/2017.
  vendor_id       : [0][0][0][0]
  encoder         : Lavc58.134.100 libvorbis
frame= 240 fps= 30 q=0.0 size=      803kB time=00:00:07.96 bitrate= 825.5kbits/s speed=0.988x
```

Y este video aparecerá en internet poniendo nuestra IP y la ruta al stream  
3.224.93.125:8000/stream.webm

## Notas:

Para poder emitir un video, este necesita previamente estar descargado en el sistema.

## 4. Ancho de banda

Para revisar el ancho de banda tenemos que instalar un paquete, por tanto, iniciaremos con un update



```
ubuntu@ip-172-31-47-116:~$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [127 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [2591 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1208 kB]
Get:7 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [2346 kB]
Get:8 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [974 kB]
Fetched 7502 kB in 1s (5397 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
30 packages can be upgraded. Run 'apt list --upgradable' to see them.
ubuntu@ip-172-31-47-116:~$
```

Y acto seguido instalar el siguiente paquete

```
ubuntu@ip-172-31-47-116:~$ sudo apt install -y iperf3
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
iperf3 is already the newest version (3.9-1+deb11u1build0.22.04.1).
0 upgraded, 0 newly installed, 0 to remove and 30 not upgraded.
```

## 4.1 Comprobaciones

Para realizar las comprobaciones, en el servidor ejecutaremos el siguiente comando, nos tendría que dar un resultado similar a este

```
ubuntu@ip-172-31-47-116:~$ iperf3 -s
-----
Server listening on 5201
-----
█
```

Desde el cliente, lo mismo, ejecutamos el siguiente comando y la salida debería de ser similar a esta

```
[ec2-user@ip-172-31-37-29 ~]$ iperf3 -c 172.31.47.116
Connecting to host 172.31.47.116, port 5201
[ 4] local 172.31.37.29 port 52894 connected to 172.31.47.116 port 5201
[ ID] Interval           Transfer     Bandwidth   Retr    Cwnd
[ 4]  0.00-1.00   sec    120 MBytes  1.00 Gbits/sec    0   717 KBytes
[ 4]  1.00-2.00   sec    122 MBytes  1.02 Gbits/sec    0   830 KBytes
[ 4]  2.00-3.00   sec    121 MBytes  1.01 Gbits/sec    0   918 KBytes
[ 4]  3.00-4.00   sec    122 MBytes  1.02 Gbits/sec    0   961 KBytes
[ 4]  4.00-5.00   sec    122 MBytes  1.02 Gbits/sec    0  1014 KBytes
[ 4]  5.00-6.00   sec    121 MBytes  1.02 Gbits/sec    0   1.11 MBytes
[ 4]  6.00-7.00   sec    121 MBytes  1.02 Gbits/sec    0   1.11 MBytes
[ 4]  7.00-8.00   sec    121 MBytes  1.02 Gbits/sec    0   1.98 MBytes
[ 4]  8.00-9.00   sec    121 MBytes  1.02 Gbits/sec    0   1.98 MBytes
[ 4]  9.00-10.00  sec    121 MBytes  1.02 Gbits/sec    0   1.98 MBytes
-----
[ ID] Interval           Transfer     Bandwidth   Retr
[ 4]  0.00-10.00  sec    1.18 GBytes  1.02 Gbits/sec    0
[ 4]  0.00-10.00  sec    1.18 GBytes  1.01 Gbits/sec
iperf Done.
[ec2-user@ip-172-31-37-29 ~]$
```

Otras opciones

Podemos medirlo solamente durante 10 segundos añadiendo -t 10 (o los segundos que necesitemos)

```

iperf Done.
[ec2-user@ip-172-31-37-29 ~]$ iperf3 -c 3.224.93.125 -t 10
Connecting to host 3.224.93.125, port 5201
[ 4] local 172.31.37.29 port 36520 connected to 3.224.93.125 port 5201
[ ID] Interval           Transfer             Bandwidth          Retr   Cwnd
[ 4]  0.00-1.00      sec    99.5 MBytes       834 Mbits/sec      39    216 KBytes
[ 4]  1.00-2.00      sec   107 MBytes       901 Mbits/sec      51    171 KBytes
[ 4]  2.00-3.00      sec   106 MBytes       891 Mbits/sec      38    202 KBytes
[ 4]  3.00-4.00      sec   106 MBytes       890 Mbits/sec      61    206 KBytes
[ 4]  4.00-5.00      sec   109 MBytes       912 Mbits/sec      25    212 KBytes
[ 4]  5.00-6.00      sec   102 MBytes       857 Mbits/sec      51    158 KBytes
[ 4]  6.00-7.00      sec   103 MBytes       863 Mbits/sec      45    215 KBytes
[ 4]  7.00-8.00      sec   108 MBytes       907 Mbits/sec      21    221 KBytes
[ 4]  8.00-9.00      sec   110 MBytes       925 Mbits/sec      54    206 KBytes
[ 4]  9.00-10.00     sec   108 MBytes       903 Mbits/sec      36    211 KBytes
-----
[ ID] Interval           Transfer             Bandwidth          Retr
[ 4]  0.00-10.00     sec    1.03 GBytes       888 Mbits/sec      421
[ 4]  0.00-10.00     sec    1.03 GBytes       888 Mbits/sec
sender
receiver

iperf Done.
[ec2-user@ip-172-31-37-29 ~]$

```

Y para medir el UDP, podemos añadir los siguientes parámetros

```

[ec2-user@ip-172-31-37-29 ~]$ iperf3 -c 3.224.93.125 -u -b 5M
Connecting to host 3.224.93.125, port 5201
[ 4] local 172.31.37.29 port 52507 connected to 3.224.93.125 port 5201
[ ID] Interval           Transfer             Bandwidth          Total Datagrams
[ 4]  0.00-1.00      sec    550 KBytes       4.51 Mbits/sec      389
[ 4]  1.00-2.00      sec    611 KBytes       5.00 Mbits/sec      432
[ 4]  2.00-3.00      sec    611 KBytes       5.01 Mbits/sec      432
[ 4]  3.00-4.00      sec    609 KBytes       4.99 Mbits/sec      431
[ 4]  4.00-5.00      sec    611 KBytes       5.00 Mbits/sec      432
[ 4]  5.00-6.00      sec    609 KBytes       4.99 Mbits/sec      431
[ 4]  6.00-7.00      sec    611 KBytes       5.00 Mbits/sec      432
[ 4]  7.00-8.00      sec    611 KBytes       5.00 Mbits/sec      432
[ 4]  8.00-9.00      sec    609 KBytes       4.99 Mbits/sec      431
[ 4]  9.00-10.00     sec    611 KBytes       5.00 Mbits/sec      432
-----
[ ID] Interval           Transfer             Bandwidth          Jitter    Lost/Total
Datagrams
[ 4]  0.00-10.00     sec    5.90 MBytes       4.95 Mbits/sec      0.024 ms  141/4274 (
3.3%)
[ 4] Sent 4274 datagrams

iperf Done.
[ec2-user@ip-172-31-37-29 ~]$

```

NOTA: Habilitar los puertos en el security group de AWS

## 5. Scripts

En este caso he creado una serie de scripts para facilitar lo que serían los trabajos de grabación de audio y emisión de video, para la generación del portal web, he creado el siguiente script

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

```
WEB_DIR="/usr/share/icecast2/web"
```

```
OUTPUT="$WEB_DIR/index.html"
```

```
# Encabezado HTML
```

```
cat > "$OUTPUT" <<EOF
```

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8" />
  <title>Portal de Audio y Video Iccast</title>
  <style>
    body { font-family: Arial, sans-serif; background: #111; color: #eee; padding: 20px; max-width:
700px; margin: auto; }
    h1 { color: #4caf50; text-align: center; }
    li { margin-bottom: 15px; background: #222; padding: 10px; border-radius: 8px; list-style:
none; }
    a { color: #81c784; font-weight: bold; text-decoration: none; }
    audio, video { width: 100%; margin-top: 5px; }
  </style>
</head>
<body>
  <h1>Archivos de audio disponibles</h1>
  <ul>
EOF
```

```
# Añadir cada archivo de audio
for file in "$(WEB_DIR)/*.{mp3,ogg,wav}"; do
  [ -e "$file" ] || continue
  filename=$(basename "$file")
  echo "  <li><a href=\"$filename\">$filename</a><br/><audio controls><source
src=\"$filename\"></audio></li>" >> "$OUTPUT"
done
```

```
# Añadir reproductor de stream de video
cat >> "$OUTPUT" <<EOF
</ul>
```

```
<h1>Stream de video en vivo</h1>
<ul>
  <li>
    <a href="http://3.224.93.125:8000/stream.webm">Ver stream.webm</a><br/>
    <video controls autoplay>
      <source src="http://3.224.93.125:8000/stream.webm" type="video/webm">
      Tu navegador no soporta video HTML5.
    </video>
  </li>
</ul>
</body>
</html>
EOF
```

```
echo "✅ Archivo HTML generado en $OUTPUT"
```

Lo que hace es listar todos los archivos de audios que haya en el directorio de web, y ponerlos en la página web, además de abajo crear una zona, en la que se emitirá el video en stream

Para el siguiente script, básicamente es un listado de todos los videos disponibles para emitir, y te pide introducir un número para emitir el video correspondiente

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



```
VIDEO_DIR="/etc/icecast2/web/video"

# Comprobar si el directorio existe
if [ ! -d "$VIDEO_DIR" ]; then
    echo "❌ El directorio '$VIDEO_DIR' no existe. Abortando."
    exit 1
fi

# Obtener la lista de archivos de video
FILES=("$VIDEO_DIR"/*)
if [ ${#FILES[@]} -eq 0 ]; then
    echo "❌ No se encontraron archivos en '$VIDEO_DIR'. Abortando."
    exit 1
fi

echo "📄 Lista de videos disponibles:"
for i in "${!FILES[@]}"; do
    FILE_NAME=$(basename "${FILES[$i]}")
    echo "$((i+1)). $FILE_NAME"
done

# Preguntar al usuario por un número de la lista
read -p "🎬 Elige un número de la lista: " SELECTION

# Comprobar si el número es válido
if ! [[ "$SELECTION" =~ ^[0-9]+$ ]] || [ "$SELECTION" -lt 1 ] || [ "$SELECTION" -gt
"${#FILES[@]}" ]; then
    echo "❌ Selección no válida. Abortando."
    exit 1
fi

VIDEO_PATH="${FILES[$((SELECTION-1))]}"

# Ejecutar ffmpeg para hacer el streaming
echo "🚀 Iniciando transmisión del video '$VIDEO_PATH' a Icecast..."
ffmpeg -re -i "$VIDEO_PATH" \
    -c:v libvpx -b:v 800k \
    -c:a libvorbis -b:a 128k \
    -f webm -content_type video/webm \
    icecast://source:pirineus@3.224.93.125:8000/stream.webm
```

Ejemplo de ejecución:

```
ubuntu@ip-172-31-47-116:/etc/icecast2/web$ ./video.sh
📄 Lista de videos disponibles:
1. concierto.mp4
2. gato.mp4
3. pan.mp4
4. pangirando.mp4
5. rickroll.mp4
🎬 Elige un número de la lista: 
```

```
ubuntu@ip-172-31-47-116:/etc/icecast2/web$ ./video.sh
📄 Lista de videos disponibles:
1. concierto.mp4
2. gato.mp4
3. pan.mp4
4. pangirando.mp4
5. rickroll.mp4
🎬 Elige un número de la lista: 6
❌ Selección no válida. Abortando.
ubuntu@ip-172-31-47-116:/etc/icecast2/web$
```

```
ubuntu@ip-172-31-47-116:/etc/icecast2/web$ ./video.sh
📄 Lista de videos disponibles:
1. concierto.mp4
2. gato.mp4
3. pan.mp4
4. pangirando.mp4
5. rickroll.mp4
🎬 Elige un número de la lista: 2
🚀 Iniciando transmisión del video '/etc/icecast2/web/video/gato.mp4' a Icecast...
```

Para los scripts de icecast, son dos scripts simples, el primero para iniciar el servicio de darkice

```
ubuntu@ip-172-31-47-116:~$ cat iniciar_darkice.sh
#!/bin/bash
#Script para configurar la tarjeta de audio

sudo modprobe snd-aloop
sudo darkice
ubuntu@ip-172-31-47-116:~$
```

Y el segundo script para grabar, guardar y enlazar el audio que queramos nosotros generar

```
ubuntu@ip-172-31-47-116:~$ cat grabar_audio_darkice.sh
#!/bin/bash
#Script para grabar con darkice

# Solicitar el nombre del archivo
read -p "Introduce el nombre del archivo de audio (sin extensión): " nombre

# Agregar extensión .mp3 si no está incluida
archivo="$nombre.mp3"

# Ruta de destino
ruta_origen="/etc/icecast2/web/$archivo"
ruta_enlace="/usr/share/icecast2/web/$archivo"

# Grabar audio
echo "Grabando audio. Pulsa Ctrl+C para detener la grabación..."
sudo arecord -D default -f cd "$ruta_origen"

# Verificar si la grabación fue exitosa
if [ -f "$ruta_origen" ]; then
    echo "Grabación completada. Creando enlace simbólico..."
    sudo ln -sf "$ruta_origen" "$ruta_enlace"
    echo "Enlace creado: $ruta_enlace -> $ruta_origen"
else
    echo "Error: no se encontró el archivo grabado en $ruta_origen"
fi
ubuntu@ip-172-31-47-116:~$
```