PROBLEMA DE KURENTO CORRIENDO EN INSTANCIA DE FIWARE LAB

En el presente documento se detalla el proceso de creación y configuración de una instancia en fiware lab con kurento, pruebas realizadas, así como los errores arrojados. Cabe mencionar que se crearon instancias y realizaron pruebas en los nodos México y España2.

Nota1: si no se especifica la instancia en la cual se realizó una configuración, instalación o prueba, dicha acción fue realizada en ambas.

Server local: Es importante mencionar que tenemos un servidor corriendo una aplicación basada en kurento player corriendo 4 cámaras y corre adecuadamente. Así como una maquina virtual simulando este servidor que igual corre adecuadamente.

PROBLEMATICA

Al correr la aplicación kurento-player cada determinado tiempo (es menor a dos minutos) aparece el WARN que se muestra en la imagen 1.

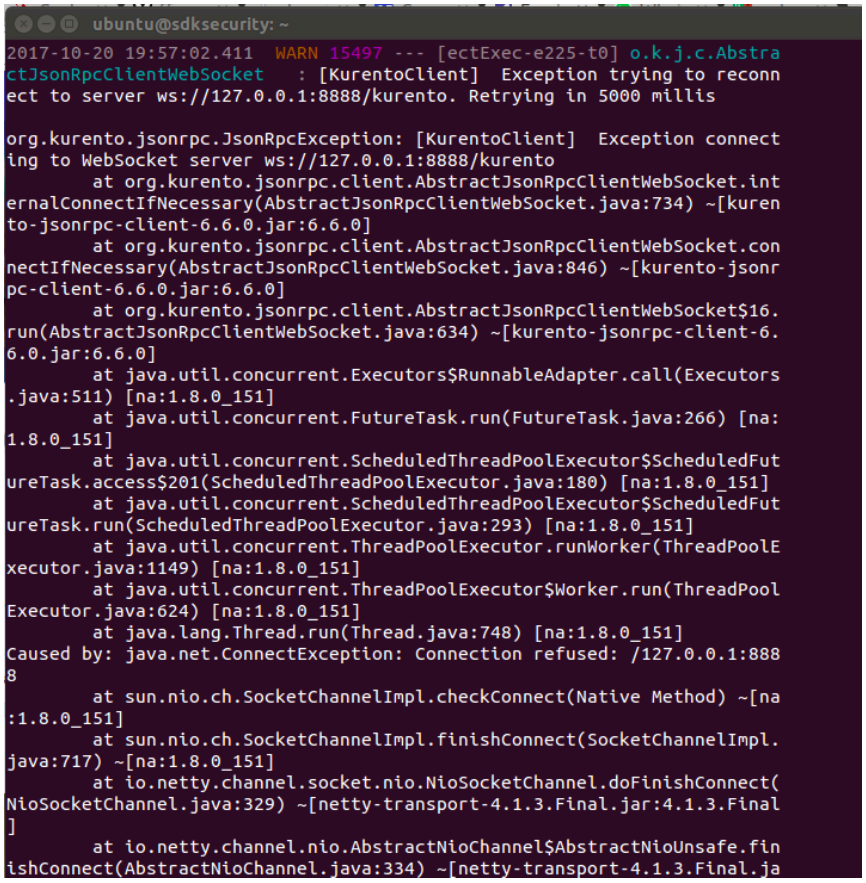

Imagen 1.- log de la consola al correr la aplicación kurento-player.

Este WARN afecta al recibir un streaming ya sea por los protocolos http, rtsp o con un vídeo que se encuentro dentro de la instancia, ya que detiene el streaming. Sin embargo dentro la aplicación web se puede presionar el botón start y da streaming por unos segundos y aparece de nuevo el error.

Nota2: algo que se observó es que la aplicación tarda menos de 100 segundos en iniciar en el nodo de España2, sin embargo, en la instancia que se encuentra en el nodo México ronda entre los 180 y 200 segundos, siendo que esta última tiene más recursos (Imagen 2).

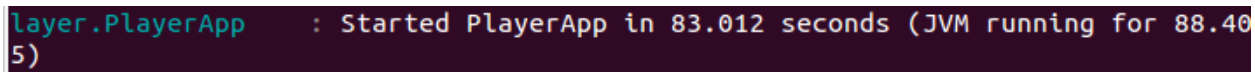

Imagen 2.- captura de pantalla del log donde se muestra cuanto tarda en iniciar la aplicación kurento player dentro de la instancia del nodo España2.

Nota3: Hemos observado que se detiene en cierta línea (Imagen 3) para ejecutar completamente aquí tarda más de un minuto (depende de la instancia que se ocupe, como se menciona anteriormente).

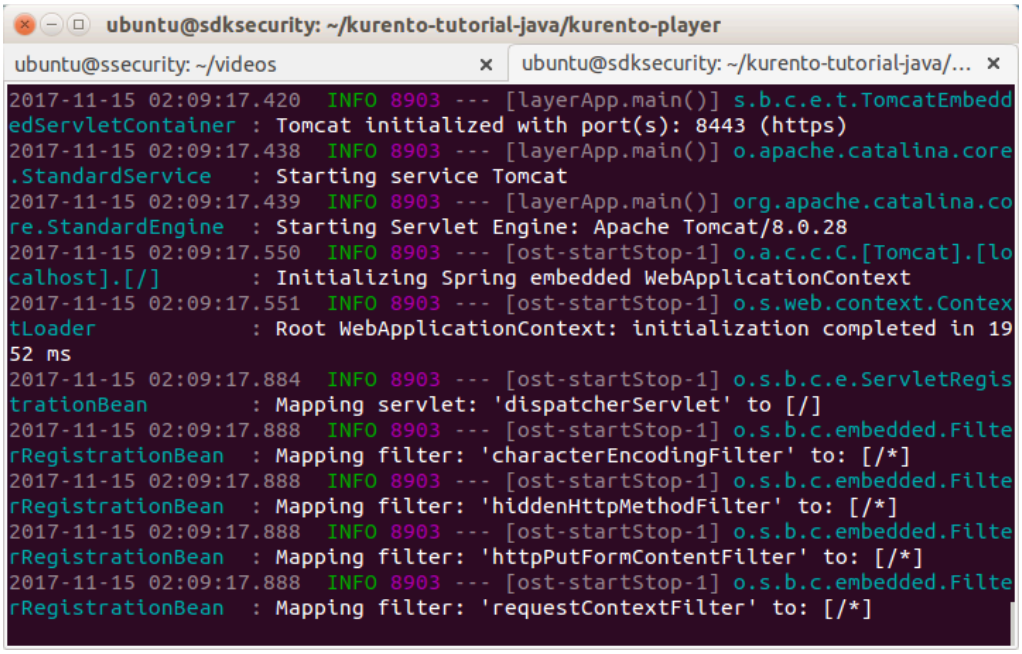

Imagen 3.- log donde se muestra en qué línea se detiene al ejecutar la aplicación kurento player

Versión del tutorial ejecutado:

Primeramente se descargaron los tutoriales de kurento en su versión 6.6.0 con los comandos que se muestran a continuación

```
git
clone https://github.com/Kurento/kurento-tutorial-java.git
cd
kurento-tutorial-java/kurento-player
git
checkout 6.6.0-SNAPSHOT
```

Como puede observarse se trabajó en el ejemplo kurento-player descrito en los tutoriales de Kurento.

CONFIGURACIÓN DE LA INSTANCIA

1.- La creación de la instancia dentro de la plataforma fiware lab fue realizada siguiendo el siguiente tutorial (https://www.youtube.com/watch?v=mb7CdMnZh7Q). Se utilizó la imagen Stream-oriented-kurento-6.6.0 (proporcionada por la plataforma fiware lab).

Recursos utilizados en la instancia del nodo México:
CPUs: 10
RAM: 30.720gb
IP: 207.249.127.160

Recursos utilizados en la instancia del nodo España2:
CPUs: 4
RAM: 8.192gb
IP: 130.206.113.226

2.- Tras crear la instancia se instalaron los siguientes programas:
Java 8
Maven 3.0.5
Git

Nota 4: normalmente al crear una instancia con la imagen Stream-oriented-kurento-6.6.0 hay un problema al usar el comando sudo, esto se resuelve entrando al archivo /etc/Hosts, agregar la direción 127.0.1.1 <nombre de usuario> (Imagen 1)

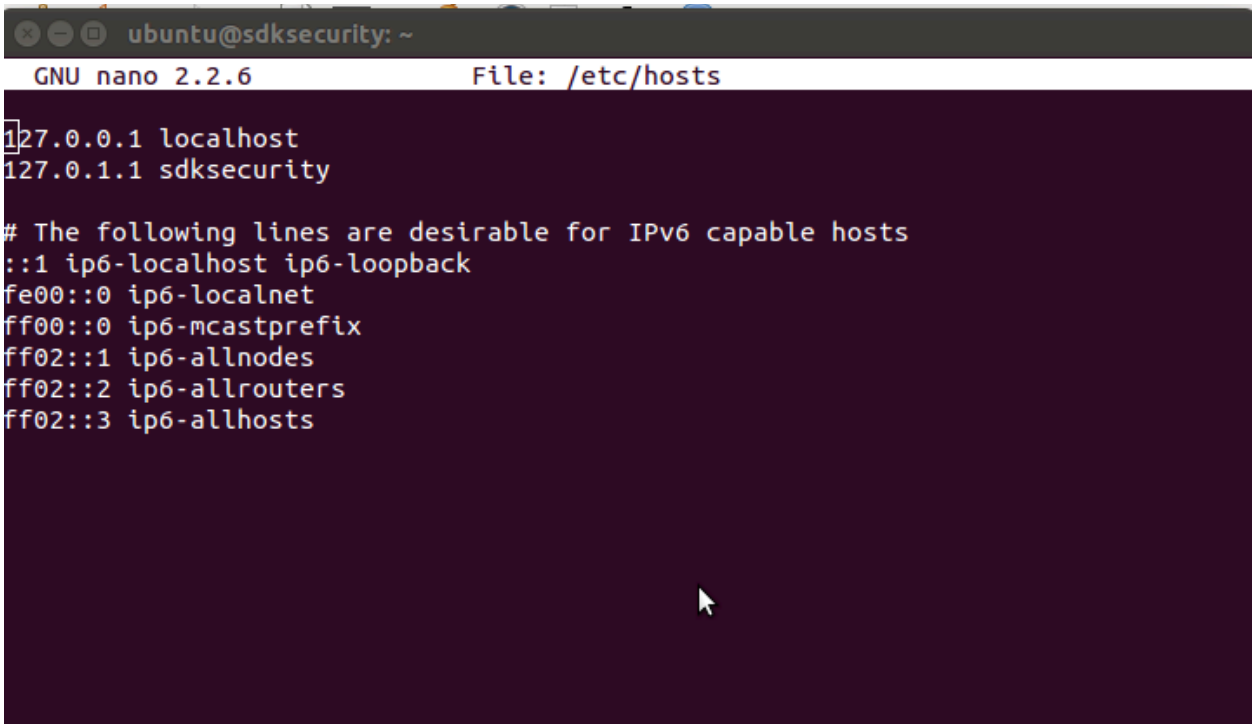

Imagen 4.- archivo de configuración de los super usuarios.

Para evitar algún problema con los puertos dentro de la plataforma de fiware lab se creó un grupo abriendo todos los puertos con todos los protocolos (Imagen 5).

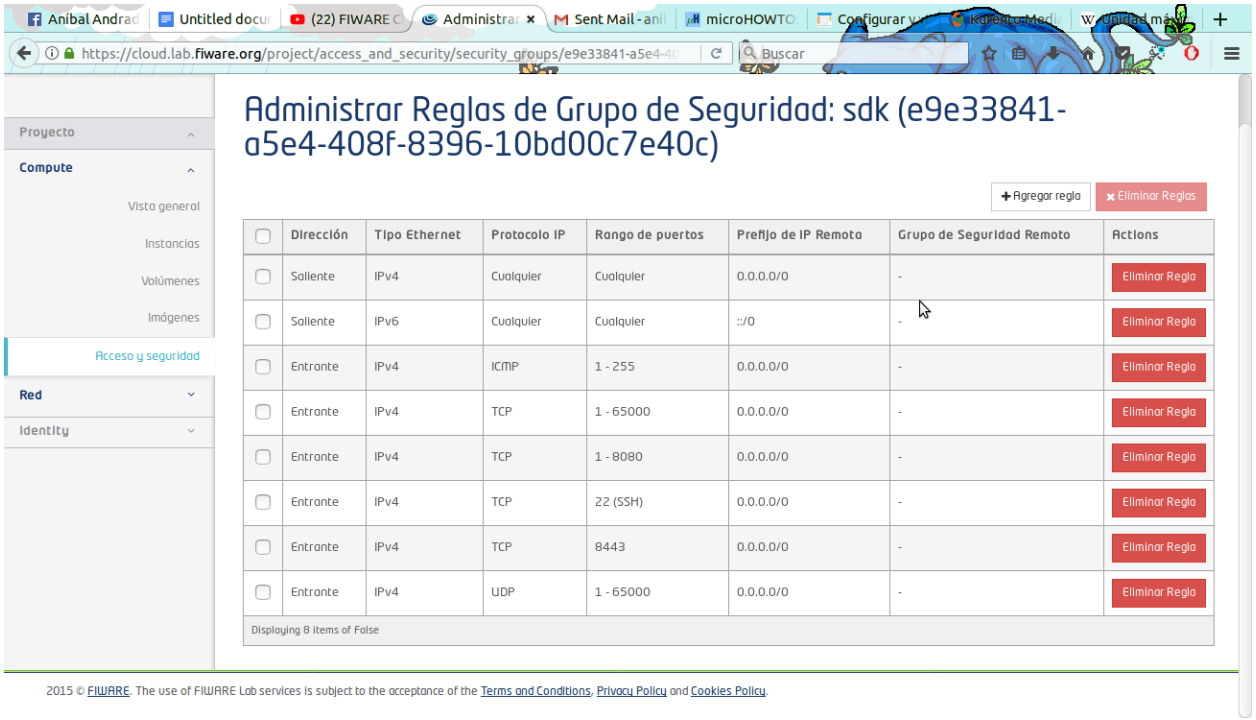

Imagen 5.- reglas del grupo de seguridad creado para la instancia del nodo en el nodo México.

Configuración Stun

Al crear la instancia con la imagen Stream-oriented-kurento-6.6.0, Kurento Media Server tiene una configuración Stun predeterminada (Imagen 6). En la instancia que se encuentra en el nodo España2 tiene el mismo puerto y la IP de la misma instancia (España2).

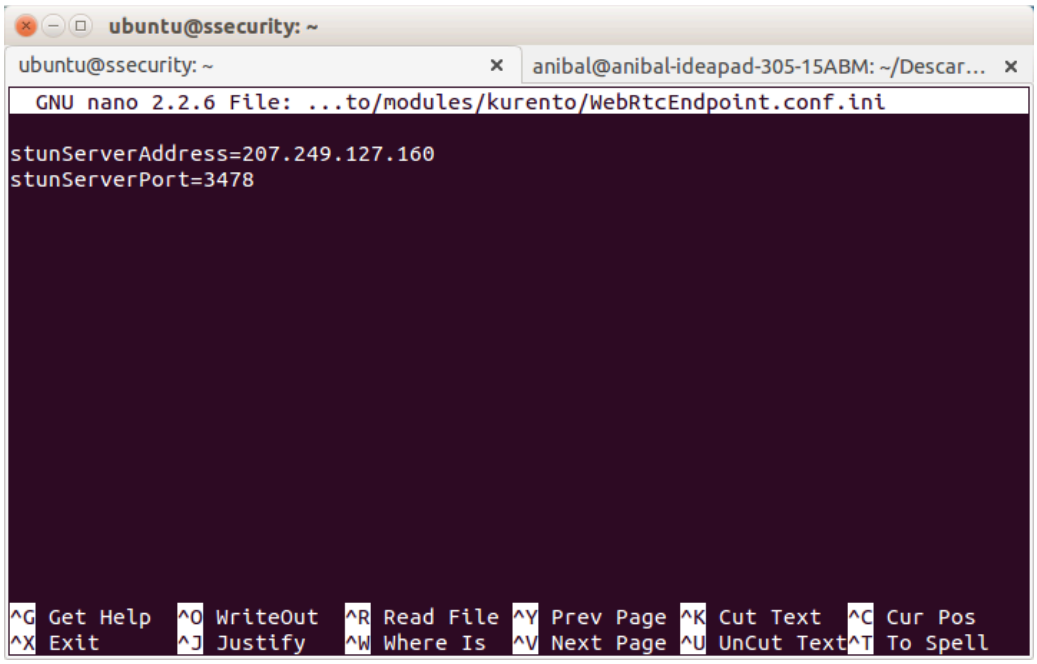

Imagen 6.- configuración Stun

Pruebas

Prueba 1: Extensión de MTU

Para esta prueba se revisó el MTU (unidad máxima de transferencia) de la instancia con el comando `ifconfig`. En la instancia del nodo México lanzó un MTU de 1454 bytes y en la instancia del nodo España2 fue 1400 bytes (Imagen 7), siendo que en cualquier equipo la interfaz eth0 tiene 1500 bytes.

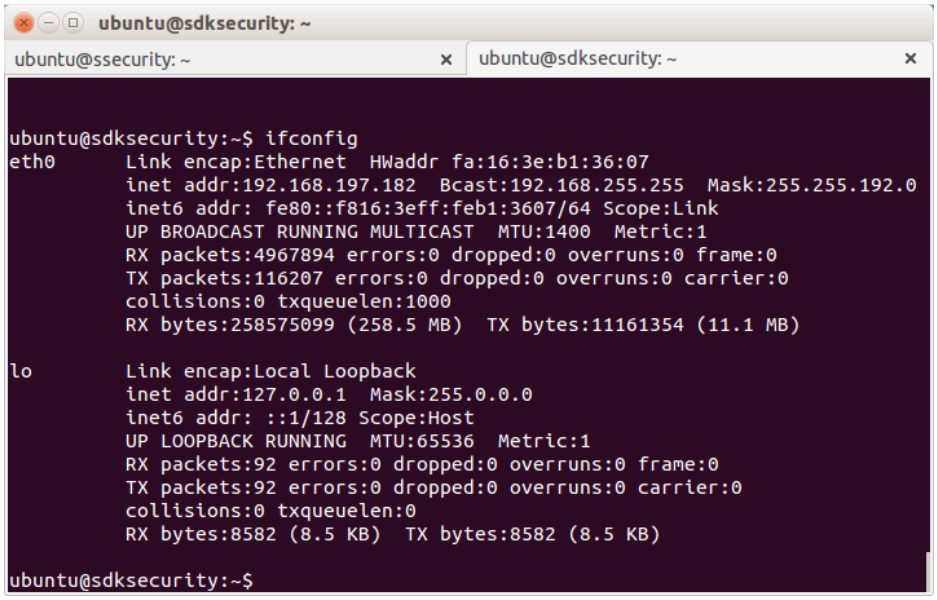

Imagen 7.- captura de pantalla que muestra el MTU dentro del nodo España2.

Para cerciorar que el WARN que arroja la aplicación (Imagen 3), si es o no dado el tamaño del MTU, osea que esté sobrepasando los bytes de transferencia se aplicó el comando `ifconfig eth0 mtu 1500` para cambiar el mtu de 1454 bytes a 1500 bytes dentro del nodo México, sin embargo esto no tuvo ningún efecto ya que al correr la aplicacion siguió mandando el mismo log.

Prueba 2: Posibles problemas con firewalls

Se probó el tener un vídeo dentro de la instancia, para descartar que fuera algún firewall que impida la conexión por más tiempo sin embargo ocurrió el mismo error. (Imagen 3).

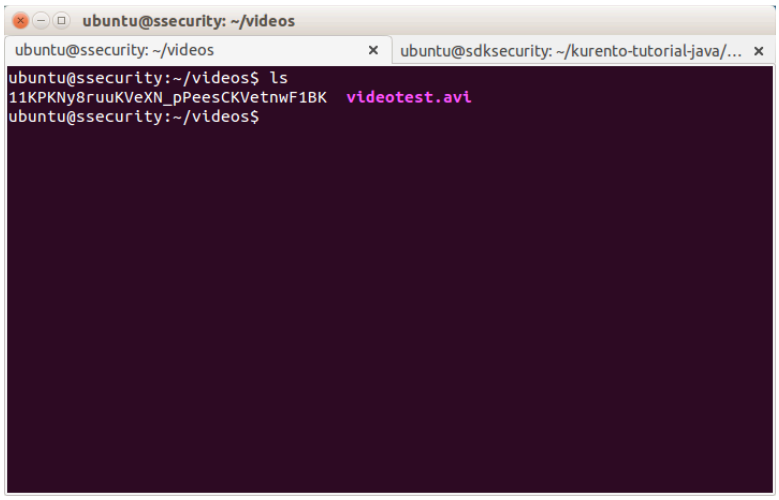

Imagen 6.- ubicación del video .avi utilizado para la prueba2

Prueba 3: Posible problemas con firewalls (creando un servidor rtp)

Esta prueba se realizó para descartar que fuera alguna conexión externa o fireware que esté impidiendo el streaming continuo. Esta prueba se realizó en la instancia que se encuentra en el nodo México.

1.- se creó un servidor rtp con ffserver (dentro de la instancia con las siguientes configuraciones:

```
HttpPort 8090
RtspPort 5554
HttpBindAddress 0.0.0.0
MaxClients 1000
MaxBandwidth 10000
```

```
NoDaemon

<Feed feed1.ffm>
File /tmp/feed1.ffm
FileMaxSize 5M
</Feed>

<Stream test.mpeg4>
Feed feed1.ffm
Format rtp
VideoCodec mpeg4
VideoFrameRate 15
VideoBufferSize 80000
VideoBitRate 100
VideoQMin 1
VideoQMax 5
VideoSize 352x288
PreRoll 0
Noaudio
</Stream>
```

2.- se mandó un streaming al servidor rtp con el programa ffmpeg, con el siguiente comando:

```
ffmpeg rtsp://x.x.x.x/path http://localhost:8090/feed1.ffm
```

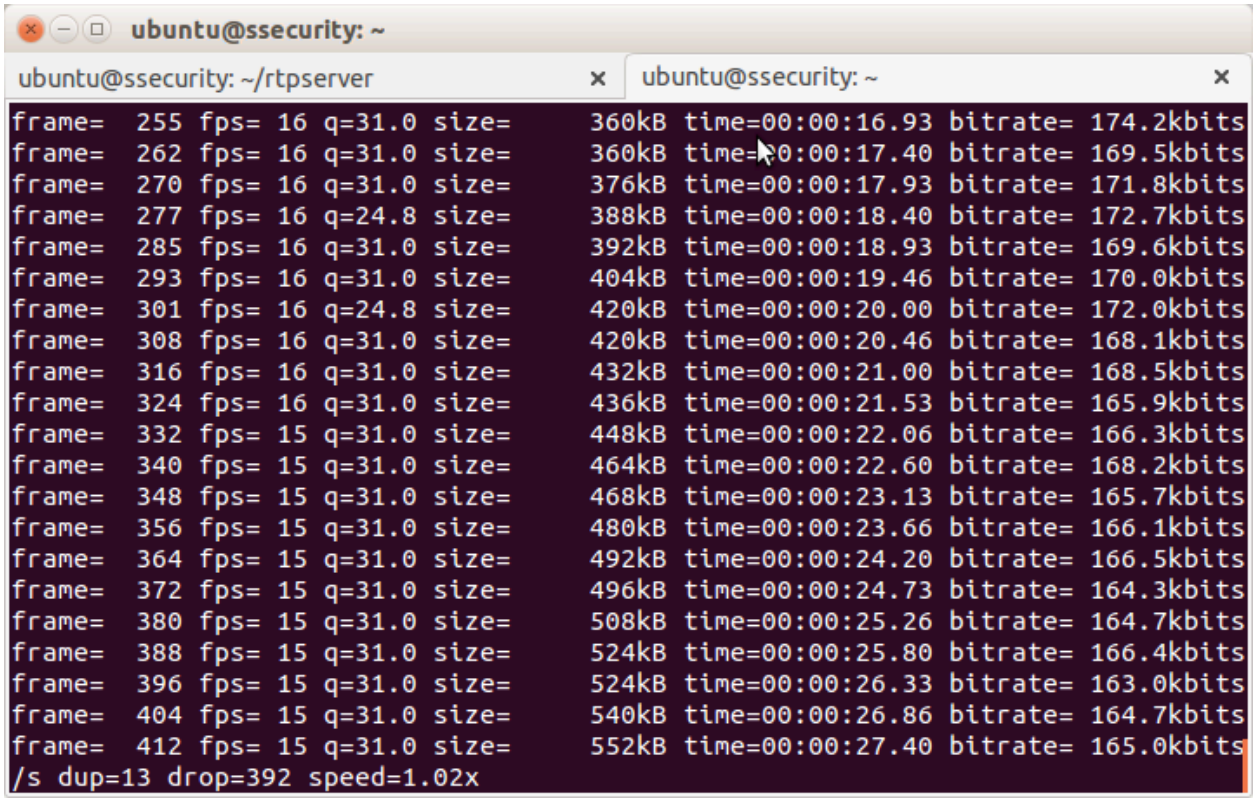

Imagen 7.- log al enviar streaming al ffserver.

3.- correr la aplicación kurento player e ingresar la dirección (http://localhost:8090/feed1.ffm) del streaming dado por el servidor rtp

El resultado fue el mismo WARN mostrado en la imagen 3.

Prueba 4: Ping

Se realizó un ping continuo desde la instancia del nodo de méxico hacia una de las cámaras utilizadas en el proyecto durante una hora teniendo un 96% de paquetes recibidos (Imagen 8).

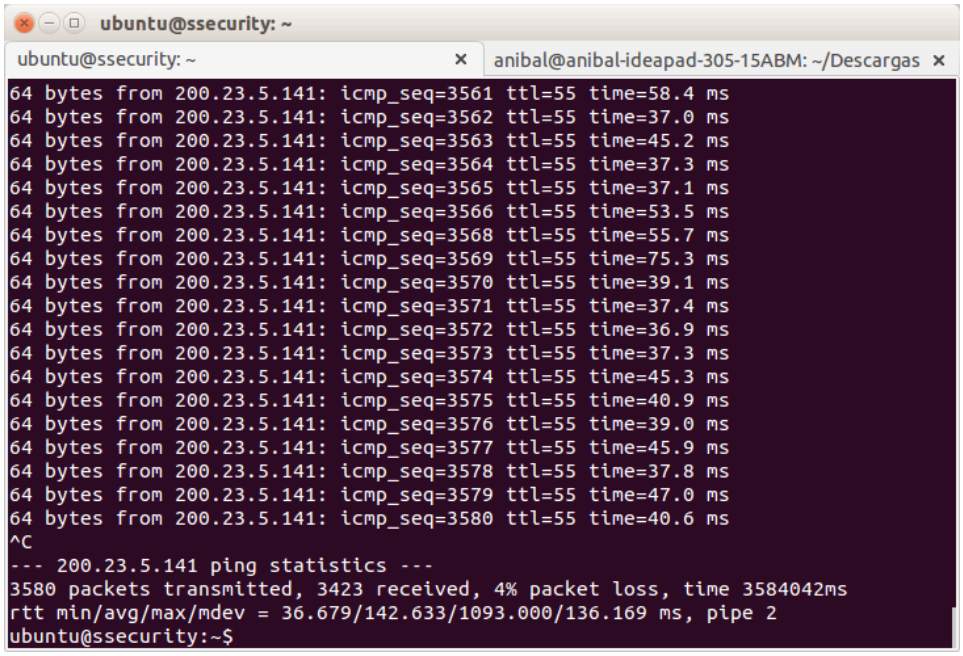

Imagen 8.- captura de pantalla ping continuo.