

Setting up FOG for Community builds



Deprecated

This document may be out of date. This feature is already included as part of RDK-6 builds.

- [Introduction](#)
- [Dependencies](#)
- [Testing FOG](#)
 - [Setting up environment](#)
- [Conclusion](#)
- [Reference](#)

Introduction

RDK-6.0 include buffering support for IP Live video playback. As of now community builds does not include fog support. This document explains how to enable the FOG support in current builds

Dependencies

FOG depends upon the storagemanager service. Storage manager service retrieves the information about all the storage devices present in the device, including SD, EMMC, NVRAM and hard drives. The type of storage device can be specified in **device.properties**. The following properties are currently used

Property	Description
SD_CARD_TSB_PART	SD Card partition for TSB
SD_CARD_MOUNT_PATH	Mount path for SD Card TSB location
EMMC_CARD_TSB_PART	eMMC TSB partition
EMMC_CARD_MOUNT_PATH	Mount path for eMMC drive

Testing FOG

FOG is for dynamic content(Live), not for static content (VOD etc) .So if the manifest contains something like

```
<MPD minBufferTime="PT1.500000S" type="static"
mediaPresentationDuration="PT0H12M14.17S" profiles="urn:mpeg:
dash:profile:isoff-live:2011">
```

it will not be buffered.

Also FOG will not be able to do trick play if the content does not have IFrames in it.

FOG requests mount path location to storage manager. At-least 1GB of free space is required for TSB to be enabled. In our test environment, we patched storage manager to return a predefined path. We also patched the fog to use a predefined location to check against free space.

Once FOG and storagemanager were up and running, we used aamp-cli to test using encoded URL.

Setting up environment

This test is done on amlogic platform. To work in different platform, export all the env variables defined in wpeframework.service on that platform

```
TOKEN=`WPEFrameworkSecurityUtility | sed -r 's/[[:",}}]/ /g' | awk '{print $2}'`
curl -X POST -H "Content-Type: application/json" -H "Authorization: Bearer $TOKEN" 'http://127.0.0.1:9998/jsonrpc' -d '{"jsonrpc": "2.0","id": 4,"method": "org.rdk.RDKShell.1.createDisplay", "params": { "client": "test-0", "displayName": "test-0" }}' ; echo

export XDG_RUNTIME_DIR=/run
export WESTEROS_GL_GRAPHICS_MAX_SIZE=1920x1080
export WESTEROS_SINK_AMLOGIC_USE_DMABUF=1
export WESTEROS_GL_USE_REFRESH_LOCK=1
export WESTEROS_SINK_USE_FREERUN=1
export WESTEROS_GL_USE_AMLOGIC_AVSYNC=1
export LD_PRELOAD=/usr/lib/libwesteros_gl.so.0.0.0
export WAYLAND_DISPLAY=test-0
```

Lauch **aamp-cli** and enter the following url

<https://cph-p2p-msl.akamaized.net/hls/live/2000341/test/master.m3u8> (HLS)

To play it as a buffered stream, it needs to be passed through FOG server. Enter this encoded form of the same URL in **aamp-cli**

http://127.0.0.1:9080/tsb?clientId=FOG_AAMP&recordedUrl=https%3A%2F%2Fcph-p2p-msl.akamaized.net%2Fhls%2Flive%2F2000341%2Ftest%2Fmaster.m3u8

Conclusion

In order to bring FOG support to accelerator platforms, we need

1. Dedicated partition (SOC/OEM)
2. Enable FOG support based on distro_feature (RDKM)

Reference

[FOG Documentation](#)

[RDKDEV-474](#)



sm-rdkstorageMain.diff



fog-downloadhelper.diff



foglogs.tgz