# AT&T Entertainment Experience Suite Video Optimizer 2.4

## User Guide

Publication Date: July 9, 2019

# Legal Disclaimer

This document and the information contained herein (collectively, the "**Information**") is provided to you (both the individual receiving this document and any legal entity on behalf of which such individual is acting) ("**You**" and "**Your**") by AT&T, on behalf of itself and its affiliates ("**AT&T**") for informational purposes only. AT&T is providing the Information to You because AT&T believes the Information may be useful to You. The Information is provided to You solely on the basis that You will be responsible for making Your own assessments of the Information and are advised to verify all representations, statements and information before using or relying upon any of the Information. Although AT&T has exercised reasonable care in providing the Information to You, AT&T does not warrant the accuracy of the Information and is not responsible for any damages arising from Your use of or reliance upon the Information. You further understand and agree that AT&T in no way represents, and You in no way rely on a belief, that AT&T is providing the Information in accordance with any standard or service (routine, customary or otherwise) related to the consulting, services, hardware or software industries.

AT&T DOES NOT WARRANT THAT THE INFORMATION IS ERROR-FREE. AT&T IS PROVIDING THE INFORMATION TO YOU "AS IS" AND "WITH ALL FAULTS." AT&T DOES NOT WARRANT, BY VIRTUE OF THIS DOCUMENT, OR BY ANY COURSE OF PERFORMANCE, COURSE OF DEALING, USAGE OF TRADE OR ANY COLLATERAL DOCUMENT HEREUNDER OR OTHERWISE, AND HEREBY EXPRESSLY DISCLAIMS, ANY REPRESENTATION OR WARRANTY OF ANY KIND WITH RESPECT TO THE INFORMATION, INCLUDING, WITHOUT LIMITATION, ANY REPRESENTATION OR WARRANTY OF DESIGN, PERFORMANCE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, OR ANY REPRESENTATION OR WARRANTY THAT THE INFORMATION IS APPLICABLE TO OR INTEROPERABLE WITH ANY SYSTEM, DATA, HARDWARE OR SOFTWARE OF ANY KIND. AT&T DISCLAIMS AND IN NO EVENT SHALL BE LIABLE FOR ANY LOSSES OR DAMAGES OF ANY KIND, WHETHER DIRECT, INDIRECT, INCIDENTAL, CONSEQUENTIAL, PUNITIVE, SPECIAL OR EXEMPLARY, INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, LOSS OF GOODWILL, COVER, TORTIOUS CONDUCT OR OTHER PECUNIARY LOSS, ARISING OUT OF OR IN ANY WAY RELATED TO THE PROVISION, NON-PROVISION, USE OR NON-USE OF THE INFORMATION, EVEN IF AT&T HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH LOSSES OR DAMAGES.

# Table of Contents

1. INTRODUCTION ..... 5

2. OVERVIEW ..... 5

2.1. INSTALLATION REQUIREMENTS ..... 5

2.1.1. VLC MEDIA PLAYER ..... 5
2.1.2. FFMPEG SETUP ..... 6

3. VIDEO OPTIMIZER TO COLLECT DATA ..... 7

3.1. PREREQUISITES ..... 8

3.1.1. IOS PREREQUISITES ..... 8
3.1.2. ANDROID PREREQUISITES ..... 9
3.2. COLLECTING A VIDEO OPTIMIZER TRACE ..... 10

3.2.1. COLLECTING A HD TRACE: IOS SPECIFICS (IOS 11 AND ABOVE ONLY) ..... 12
3.2.2. COLLECTING A SECURE TRACE (ANDROID ONLY, UP TO PIE) ..... 14
3.2.3. COLLECTING AN ATTENUATED TRACE ..... 15
3.2.4. IOS SECURE COLLECTOR ..... 19
3.3. TESTING YOUR APPLICATION ..... 23

3.4. ENDING THE TRACE ..... 23

4. ANALYZING A TRACE ..... 25

4.1. OPENING A TRACE FILE ..... 25

4.2. BEST PRACTICES RESULTS ..... 27

4.3. PRIVATE DATA TRACKING ..... 30

4.4. VIDEO ANALYSIS ..... 32

4.4.1. VIDEO ANALYSIS SETUP ..... 32
4.4.2. CONFIGURATION REQUIRED ..... 33
4.4.3. ANALYZING A VIDEO ..... 33
4.4.4. VIDEO BEST PRACTICES ..... 36

5. VIDEO OPTIMIZER REFERENCE ..... 39

5.1. MENU ..... 39

5.1.1. FILE MENU ..... 40
5.1.2. PREFERENCES ..... 41

5.1.3. PROFILE MENU ..... 44
5.1.4. TOOLS MENU ..... 52
5.1.5. VIEW MENU ..... 54
5.1.6. DATA COLLECTOR MENU ..... 58
5.1.7. HELP MENU ..... 58
5.2. CONTENT TABS ..... 59
5.2.1. BEST PRACTICES/RESULTS TAB ..... 59
5.2.2. OVERVIEW TAB ..... 69
5.2.3. DIAGNOSTIC TAB ..... 74
5.2.4. VIDEO TAB ..... 88
5.2.5. STATISTICS TAB ..... 98
5.2.6. WATERFALL TAB ..... 116

6. APPENDIX I ..... 119

6.1. DATA COLLECTOR ERROR MESSAGES ..... 119
6.2. VIDEO OPTIMIZER ERROR MESSAGES ..... 121
6.3. GLOSSARY ..... 131

7. APPENDIX II ..... 138

7.1. ROOTED DATA COLLECTOR APK ..... 138
7.1.1. PREREQUISITES FOR USING THE ROOTED DATA COLLECTOR APK ..... 138
7.1.2. INSTALLING ROOTED DATA COLLECTOR APK ..... 139

# 1. Introduction

The Video Optimizer User Guide describes in detail the methods for collecting an application trace with Video Optimizer. This guide describes how to open a trace and it includes a reference section with the tabs, menus, options, charts, graphs, and statistics. This guide is intended for app developers who are interested in testing the performance of their apps.

# 2. Overview

Video Optimizer is a diagnostic tool for analyzing mobile web application performance. Video Optimizer automatically profiles your application and provides recommendations that let you optimize performance, make battery usage more efficient, and reduce network impact.

The traces that Video Optimizer runs against your application are benchmarked against recommended best practices. Video Optimizer looks at how your application (and your server) is handling caching, and how you are managing the network connections for your application. By optimizing against these best practices, your application will run faster, use the network less (saving valuable battery life for your users), and improve the experience of customers using your application.

## 2.1. Installation Requirements

Before you download and install Video Optimizer 2.4, be sure that your computer has at least 2GB of RAM. In addition, you will need the following software installed in order to open & analyze traces:

- Java 8 or above (ie: Java 1.8)
- Wireshark (Note: Install Npcap addon with Winpcap API-compatible mode
- FFmpeg
- VLC media player (Windows & Linux only)
- Additional software is needed to use Video Optimizer to collect traces from Android or iOS devices. More info is in section 3 below. The high-level list is:
- Android SDK Level 19 or above (for Android traces)
- The latest version of XCode (for iOS traces)
- Brew (Mac/iOS)
- Ifuse & OSXFuse (Mac/iOS)
- Libimobiledevice & ideviceinstall (Mac/iOS)

### 2.1.1. VLC Media Player

Video Optimizer uses VLC media player to play most types of videos (namely mp4/SD|HD and mov/low-res videos). However, there is some environment setup required for the player to work – you need to have VLC installed on your computer (you can download it from http://www.videolan.org/vlc/) and have it installed in a specific location.

#### 2.1.1.1. Windows Setup

For Windows, you need to have an environment variable setup that points to the VLC plugins folder. Perform the following steps.

1. Have VLC installed in one of the following locations:
   - C:\Program Files (x86)\VideoLAN\VLC
   - C:\Program Files\VideoLAN\VLC
2. Have an environment variable named "VLC_PLUGIN_PATH" that points to the location of the plugins folder and include it in your path. E.g.:
   - C:\Program Files\VideoLAN\VLC\plugins
   - C:\Program Files (x86)\VideoLAN\VLC\plugins

#### 2.1.1.2. Linux Setup

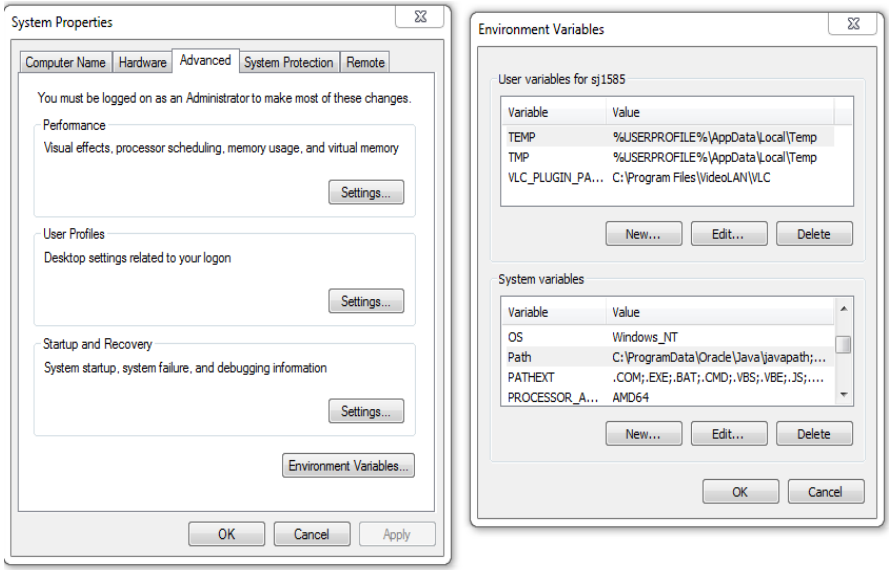
For Linux, you need to have VLC installed at the location /usr/lib/vlc

### 2.1.2. FFmpeg Setup

FFmpeg is used by Video Optimizer in a few different ways, primarily associated with Video Analysis. This too needs an environment setup before it can work correctly.

#### 2.1.2.1. Windows Setup

Navigate to set environment system variables and user can create/edit an environment label named "Path" and point to the install location for FFMPEG.




#### 2.1.2.2. Mac OS Setup

Two options to install and configure FFMPEG

1. If you already have Brew installed, you can install ffmpeg using Brew.
   - "Brew install ffmpeg" will install both FFmpeg and FFprobe on MAC.
2. Install FFmpeg, FFprobe etc. from https://www.ffmpeg.org/ and then adding a Symbolic link for the same in '/usr/local/bin'.
   - E.g. ln -s [path of ffmpeg] /usr/local/bin/ffmpeg

#### 2.1.2.3. Linux Setup

FFmpeg is available in the default repositories. Open Terminal and run the following command to install it.

- Sudo apt -get install ffmpeg

# 3. Video Optimizer to Collect Data

Video Optimizer captures the data traffic of mobile devices. As data streams across the network during a given period, Video Optimizer captures each TCP packet and matches the packet information with recorded video of what the user is seeing on the device. Video Optimizer also looks at other parameters from the mobile device such as signal strength, network type, CPU & GPS usage, etc.

You can use Video Optimizer to test the following:

- iOS device versions 6 and up.
- Android devices (and emulators) versions 4.4 and up.
- Note: The original Android Collector works on Android versions 2.3.7 - 5.1 but requires root access. See APPENDIX II for more information about how to use the legacy Rooted Data Collector API.
- PCAP Packet traces: PCAP packet trace files contain basic (network only) data that can be captured using several different tools. Once captured, PCAP files can be opened directly in the Video Optimizer.
- Automated Collection using Video Optimizer SDK. To facilitate the integration of Video Optimizer with developer tools, automated testing environments and Enterprise build environments, we've exposed APIs for collecting and analyzing traces.

## 3.1. Prerequisites

This section describes the prerequisites for collecting a Video Optimizer trace on iOS and Android devices.

### 3.1.1. iOS Prerequisites

Collecting a Video Optimizer trace on an iOS device requires the following:

Mac computer running OSX 10.8 or higher, with the following:

1. Administrator rights
2. Ability to use SUDO password from Terminal
3. Oracle Java:

   Required version 1.8(JRE 8) for analyzing traces. Video Optimizer will not work as expected with lower version of Java.
4. Apple's XCode

   Required to capture and analyze iOS traces.
5. Wireshark

   **Required** for maintaining cross-platform compatibility of traces.
6. Homebrew

   Recommended to simplify the install of the following software packages used by Video Optimizer
7. FFmpeg

   **Required** for high definition video and video analysis. If using Homebrew, install FFmpeg from the Terminal window using "brew install ffmpeg"
8. FUSE, libimobiledevice & ideviceinstall for macOS

**Required** for obtaining HD iOS traces.

Steps to follow if using BREW:

- brew update
- brew cask install osxfuse
- brew install ifuse
- brew uninstall --ignore-dependencies libimobiledevice
- brew uninstall --ignore-dependencies usbmuxd
- brew uninstall libplist
- brew install --HEAD libplist
- brew unlink libplist
- brew link libplist
- brew install --HEAD usbmuxd
- brew unlink usbmuxd
- brew link usbmuxd
- brew install --HEAD libimobiledevice

9. Android SDK

   Optional - only required if the Android traces are collected on Mac.

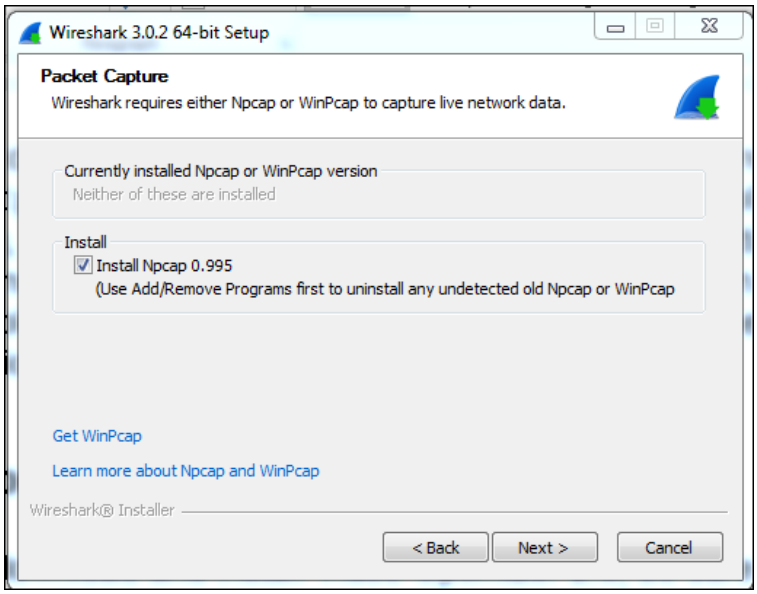
### 3.1.2. Android Prerequisites

To collect a Video Optimizer trace on an Android device, you must have the following:

1. PC, Mac or Linux computer along with Android SDK installed (for ADB control of device).
2. Android device (or emulator) running 4.4 or higher.
3. ADB path set in Preferences dialog.
4. Oracle Java

   **Required** version **1.8** for analyzing traces. Video Optimizer will not work as expected with lower version of Java.
5. WinPcap/Npcap (included in Wireshark install)

   **Required** for analyzing traces on Microsoft Windows and if missing AT&T Video Optimizer will not run. You must select the option to Install Npcap in WinPcap API-compatible mode




6. FFmpeg

   Required for high definition video and video analysis.

7. VLC media player

   Required for playing video and video analysis in Microsoft Windows.

8. Android SDK

   Included in Android Studio.

   Windows path

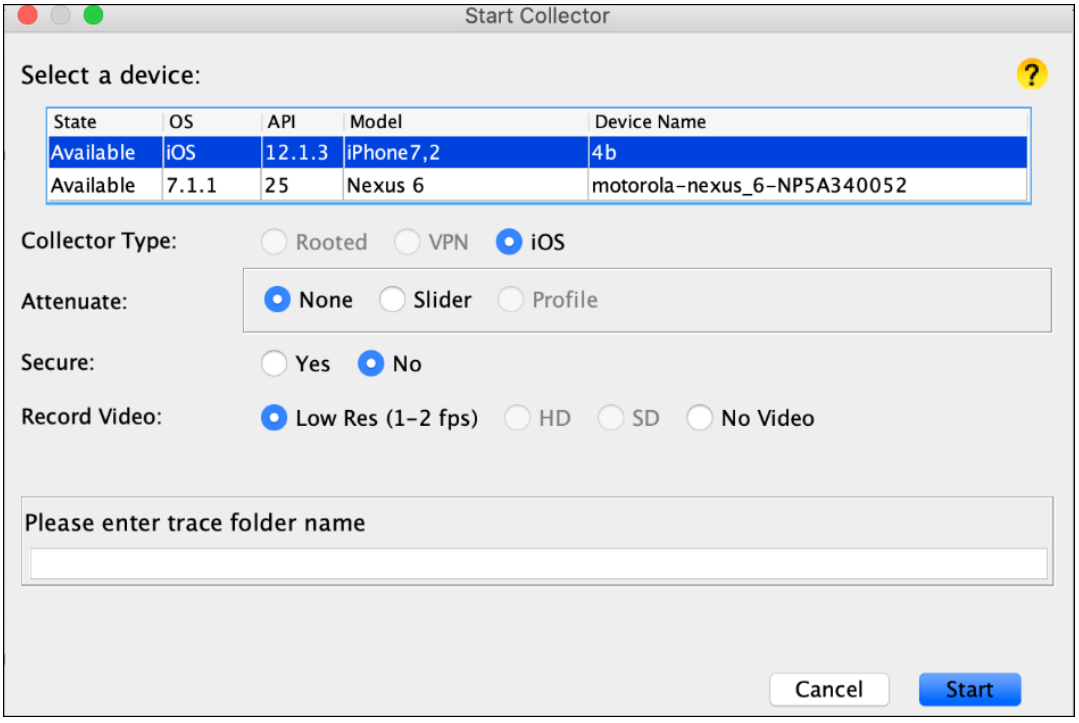
   Add the locations of these components to your windows path:

   - Start Button>Computer>System Properties>Advanced system settings>Environment Variables
   - Scroll down System variables to “Path” and select Edit
   - Add the paths to adb.exe, ffmpeg.exe and the VLC\plugins folder to the end of the existing Path.

## 3.2. Collecting a Video Optimizer Trace

Collecting a trace works similarly for all devices. Simply connect the device to your computer and open the Video Optimizer application.

Step 1. Select Start Collector from the Data Collector menu.


Step 2. Select a Device in the Start Collector dialog. Video Optimizer will discover all iOS and Android devices connected to the computer. **Figure** shows that two devices are connected: an iOS device and an Android device, with the Android device selected.

Step 3. Select the Collector Type. The recommended setting is VPN, especially for an Android device that does not have root access. Other collector types are available for Android devices that are rooted.

Step 4. ***Optional:*** Select an Attenuate option. This lets you reduce the throughput of the network connection, allowing tests on networks with lower bandwidth profiles (for more information, see the Network Attenuation section). In the **Figure** above Attenuation is not selected, so it is not being used.

Step 5. ***Optional:*** Secure collection is available on Android versions 4.4- 9. Select Yes for Secure if you want Video Optimizer to decrypt HTTPS traffic – providing you with a more thorough analysis of the traffic transmitted during your test. You will be prompted to install a certificate if this is your first time to use secure collection. On subsequent tests, you do not need to install the certificate.

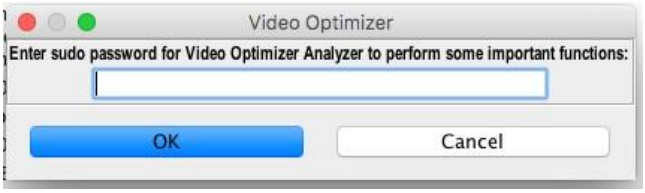

Step 6. Select a Record Video option. Depending on your device, there will be two or more options available for recording the screen while you are collecting Video Optimizer data.

- Low Res:1-2 frames per second. Advantage: Smaller video size
- HD: 8 MBPS, 1920x1080.
  ***Note:*** This will generate very large video files. Generally, you should only use this format for short traces. If HD video record fails, it may indicate that there were not enough system resources, so try with SD or Low-res settings.
- SD:SD video 3MBPS, 960x540 video – for a smaller size (but lower quality) video.

Step 7. Type a trace name in the "Please enter trace folder name" box. This is where all the trace files will be stored. Spaces and special characters are not allowed and if you pick an existing name you will be asked if you want to overwrite the previous trace.

Step 8. Click Start to begin collection on your device after you have selected the appropriate parameters or click Cancel to prevent a trace from being run.

Step 9. ***iOS Only:*** Video Optimizer will prompt you to enter the OS password, this is required to start the traffic capture utility.




## 3.2.1. Collecting a HD Trace: iOS Specifics (iOS 11 and above only)

iOS11 added the functionality to be able to record the device screen. We now have a companion app, capable of pulling the recorded video off the device as long as the user requests it. The following section details some additional steps that must be done to collect a HD video on an iOS device.

### 3.2.1.1. Enabling an iOS Device for Development

If you haven't enabled Developer settings on your iOS device, this section details how to do that. This needs to be done only once.

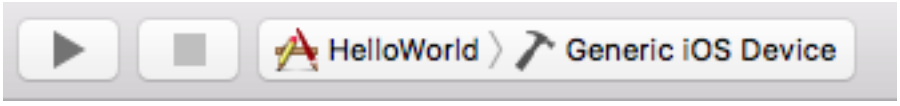
1. Connect the iOS device to the Mac computer.
2. Open XCode and select Organizer from the Window menu.

3. Select the connected device from the list of devices in the left nav.
4. Click Use for Development.
5. Click Cancel when you see a prompt to join the “Apple developer program and login with a paid developer account.” (You do not need a paid account to continue.)

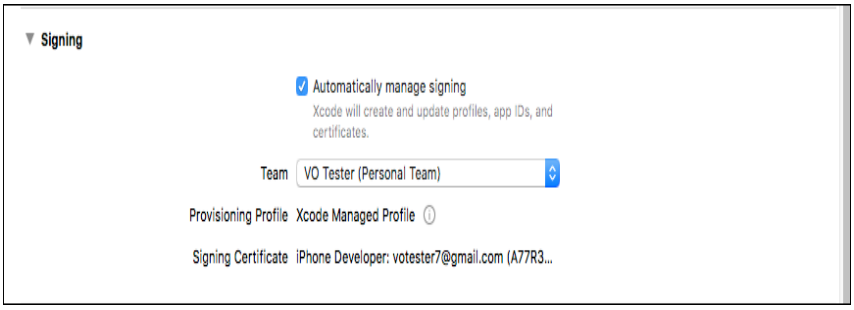
   **Note:** Always start XCode before you start Video Optimizer so that you can see that the device is detected and ready to use.
6. When the device is enabled for development, you will see a “Developer” option on the Settings menu of the device.

#### 3.2.1.2. Collect iOS HD Video

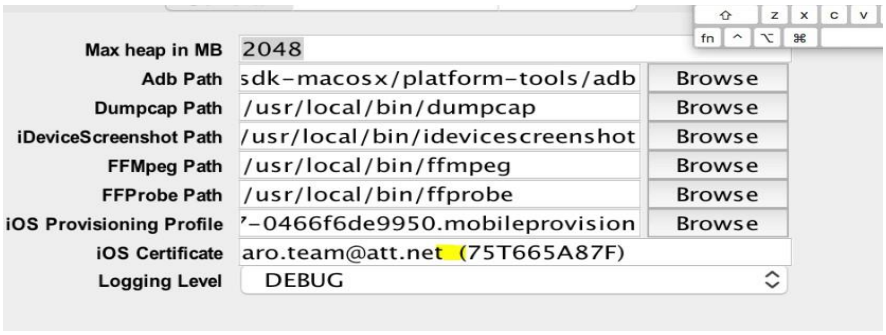
1. Attach the iOS device, that you want to collect from, to the Mac.
2. Navigate to XCode and Create a “New Project” (This is applicable if you don’t have an existing project/Signing Certificate/Provisioning Profile).
3. Select the connected device from the toolbar mapping the project to a device.




4. This should generate a provisioning profile and the developer certificate.




5. Copy the Signing Certificate Information [email [Certificate number]] and add it to Video Optimizer >> File >> Preferences >> iOS Certificate.
6. Navigate to "~/Library/MobileDevice/Provisioning Profiles/[Latest Provisioning Profile]” and add it to Video Optimizer >> File >> Preferences >> iOS Provisioning Profile.

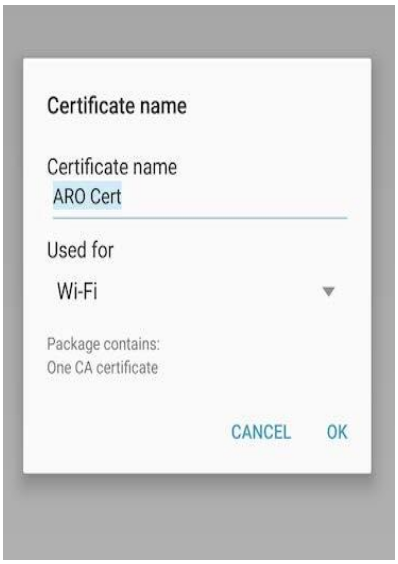



7. These settings would allow for the Video Optimizer iOS app to be installed on an iPhone running iOS 11.
8. Start a HD trace collection on iOS, and once the app is installed on the phone (If you are a developer who doesn't have any other apps running on the phone) - do the following
9. Navigate to the iPhone >> Settings >> General >> Profiles and Device Management, click on the Video Optimizer App and trust the corresponding developer.

### 3.2.2. Collecting a Secure Trace (Android only, up to Pie)

The Video Optimizer android collector can do an MITM based capture of secure traffic. This requires a user certificate installed on the phone on which the trace is being captured. Following are the steps to be followed to capture a secure trace.

Step 1. Install a certificate (required on first test)—if you have selected a secure trace and have opted to install a certificate. Choose the VPN and apps (not Wi-Fi) option for Certificate name on your device and select OK.




Step 2. Select OK to allow the VPN connection to start on the device—when the trace is started, Video Optimizer installs a collection app on the device. The app establishes a VPN connection and requests permission to allow the connection.

***Note:*** The VPN connection is a connection to nowhere. It originates and terminates on the device. All packets flow through the network normally.

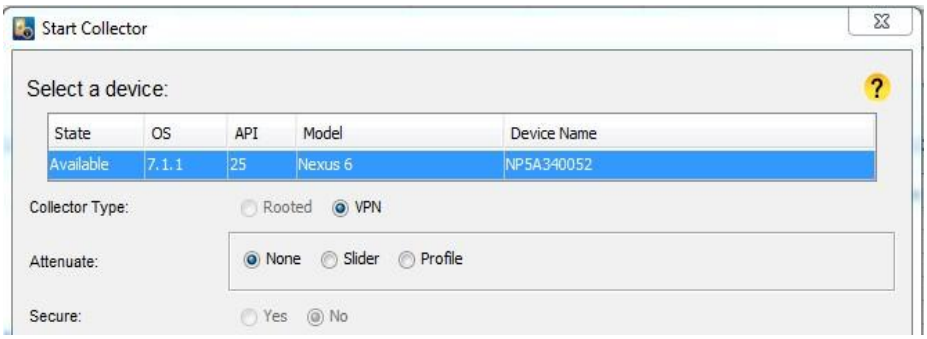
### 3.2.3. Collecting an Attenuated Trace

Video Optimizer can attenuate a trace collection, restricting the upload/download speed of traffic flowing through the VPN/MITM.

#### 3.2.3.1. Android Attenuation

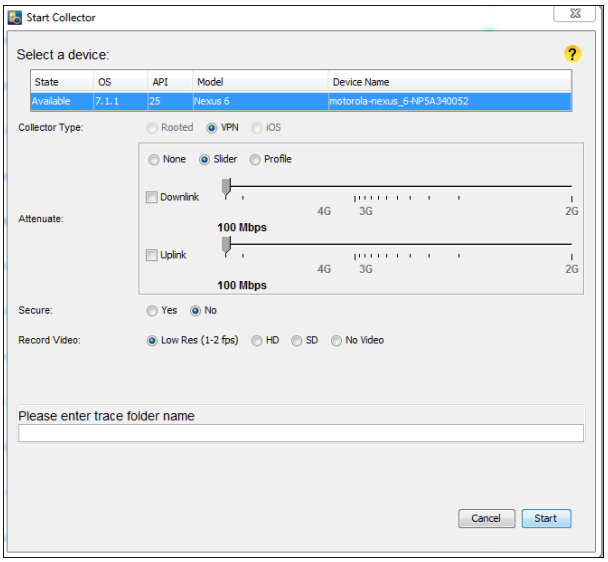
When you collect a trace, you can add a network throughput cap for uplink or downlink, set in KBPS. This will limit all data traffic on the phone to a maximum data throughput. This is a useful way to test your mobile application in a network constrained environment; perhaps at a busy sporting event, or in a country that does not yet have a LTE network.

When you are collecting a trace, the default attenuation setting is "None," meaning that no attenuation is performed by default.




#### 3.2.3.1.1. Static Attenuation

Step 1. Select Slider from the attenuate options.




Step 2. Select Downlink to lower the throughput of the packets being downloaded from the server or select Uplink to lower the throughput of the device uploading files to the server.

Step 3. Set the speed to 4G, 3G or 2G on the slider. A value between 3G and 2G will allow tests on “slow 3G.” Video Optimizer sets the attenuation value that you selected as the maximum throughput available for the entire test. Note: If you set a maximum throughput value that is higher than the actual speed of your network, you will not see an improvement. Video Optimizer attenuation will only SLOW the network throughput – it cannot enhance the speed of your network. If your phone is on a slow network, and attenuation is set to 4G, it will still be a slow connection.

#### 3.2.3.1.2. Dynamic Attenuation

The Attenuate feature in Video Optimizer is a useful way to test how apps and video behave in a constrained environment. But when you are testing mobile video, it is also interesting to consider how changes in network throughput affect a stream that is in progress.

When network changes occur while a video is playing, the player may change the bitrate of the video to ensure that the video buffer remains full – because when the buffer is dry, the video will stall. It may request multiple versions of the same segment – yet only play the highest quality segment received. Understanding how your video stream behaves in changing network environments is the best way to understand and prevent stalls from occurring.

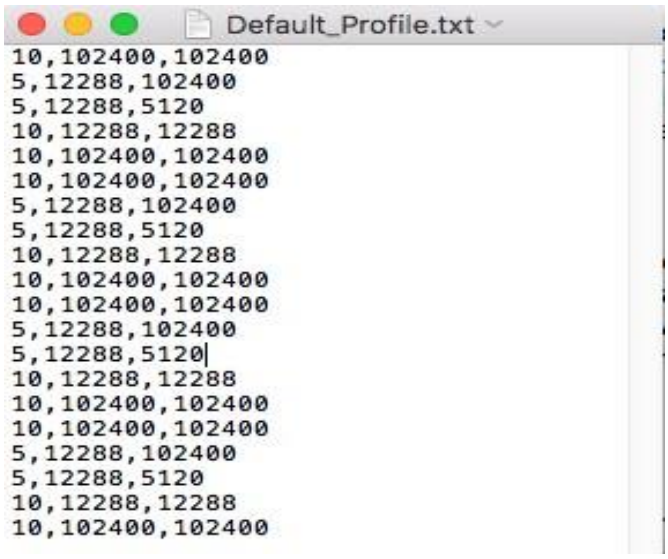
The dynamic profile option allows you to change the network speeds while the trace is running. Now you can test your app or video in changing network conditions (that you can control and re-use systematically and in a repeatable manner).

Step 1. Select Profile from the attenuate options.


Step 2. ***To modify the default profile***, you can create your own script file—just modify the default_profile.txt document in a text editor and save with a new name.

Step 3. The format of the text file is comma delimited, with 3 entries per line: time in seconds, downlink in KBPS, uplink in KBPS. 102400 KBPS indicates "no throttle" in that time frame.

Step 4. For example, in the figure below, we begin with 10 seconds of no throttle, followed by 5 seconds at 12 MBPS in downlink (no uplink throttle). Then the downlink is slowed to 5 MBPS for 5 seconds, sped up again for 10 seconds, and 30 seconds into the test, both are unthrottled again.

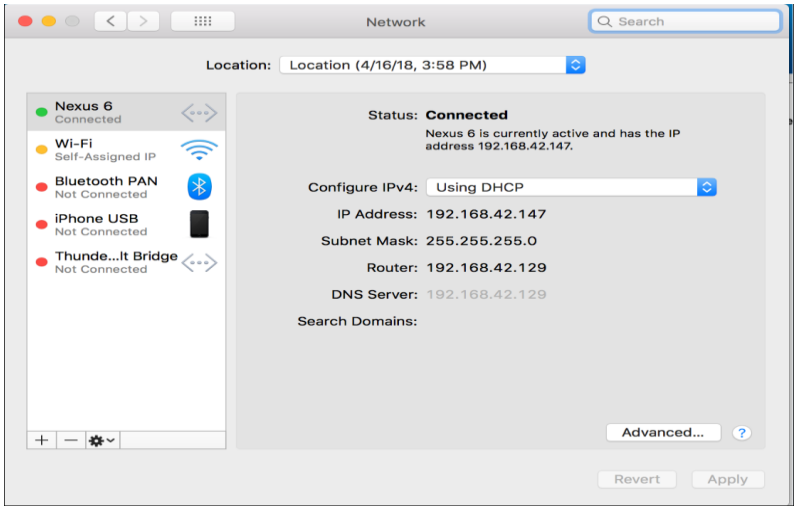



### 3.2.3.2. iOS Attenuation

**Following are the steps to setup the environment on MAC.**

Step 1. Choose Apple menu > System Preference, then click Network.

Step 2. Select Ethernet (wired) or the USB tethering for network connection.



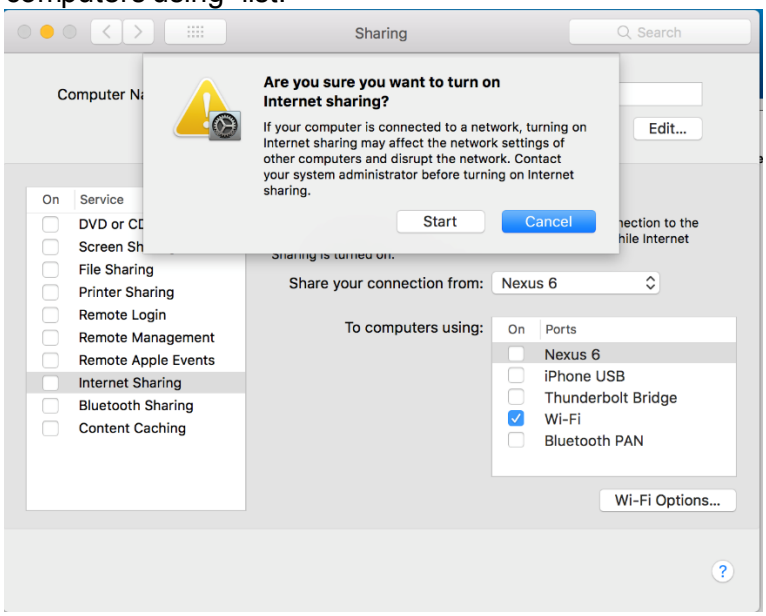

Step 3. Turn the Mac Hotspot on.

Step 4. Choose Apple menu > System preferences, then click Sharing  , open Sharing preferences if it isn't already open.

Step 5. Select the internet Sharing checkbox.

Step 6. Click the “Share your connection from” pop-up, then choose the Internet connection you want to share.

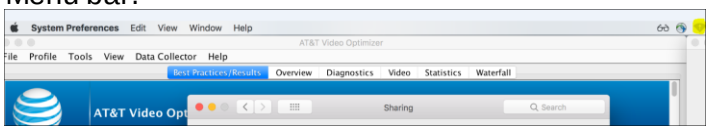
Step 7. Select the type to share your internet connection in the “To computers using” list.




Step 8. Select Wi-Fi.

Step 9. If your Internet connection and your local network use WIFI, sharing your internet connection disrupts the network.

Step 10. Observe on the Mac, the internet sharing icon is seen in the Menu bar.




**Following are the steps to setup the environment on wi-fi on iOS device:**

Step 1. Go to settings>Wi-Fi. Turn on Wi-Fi and Airplane mode.

Step 2. For wi-fi: select the Mac computer shared hot spot from the listed networks, then enter the password, if required.

Step 3. Adjust the settings for a Wi-Fi network: Select the more Info button next to a network. Set up a HTTP proxy.

Step 4. Set Proxy: Enter the IP address of the Mac computer Hotspot and port number is “8080”

Step 5. Depending on the device make sure the wi-fi and air-plane mode are enabled and displayed.

### 3.2.4. iOS Secure Collector

**Introduction:**

Video Optimizer iOS collector can be used as a man-in-the-middle HTTPS proxy, enabling you to view the communication between web browser and SSL web server.

Video Optimizer does this by becoming a man-in-the-middle. Instead of your browser seeing the server's certificate, Video Optimizer generates a certificate for the server and signs it with its own root certificate (the Video Optimizer CA Certificate). Video Optimizer receives the server's certificate, while your browser receives Video Optimizer's certificate. Therefore, you will see a security warning, indicating that the root authority is not trusted. If you add the Video Optimizer CA Certificate to your trusted certificates you will no longer see any warnings – see below for how to do this.

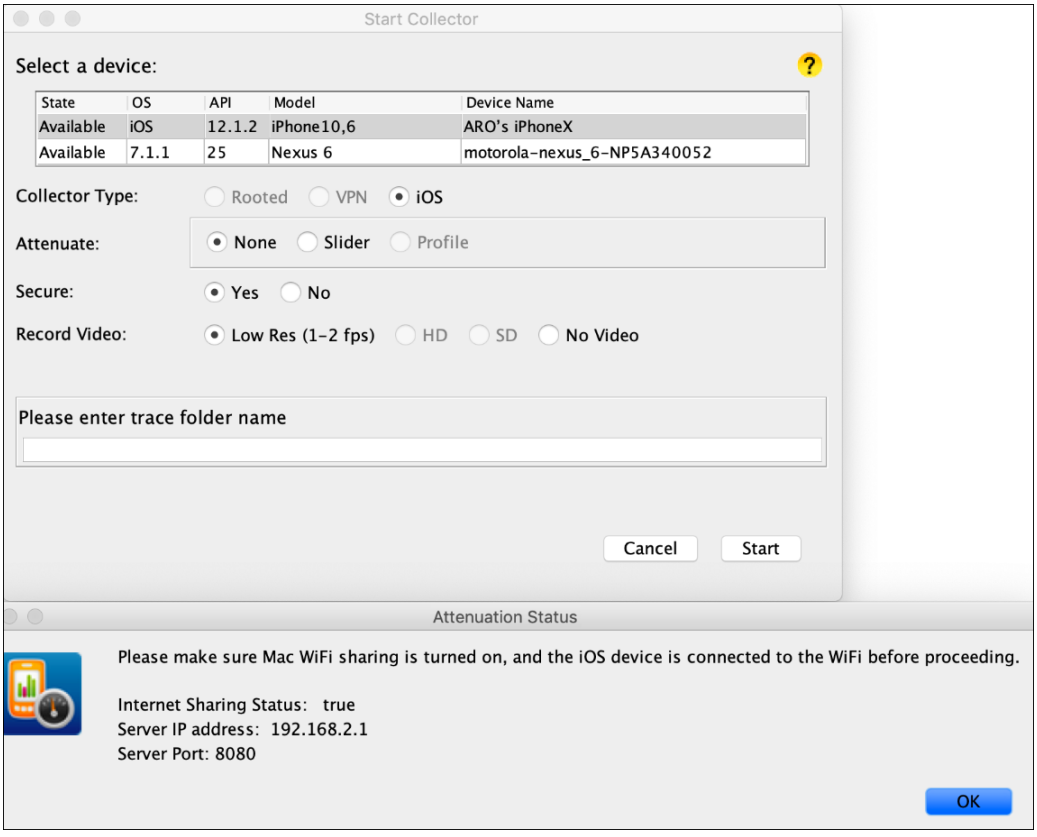
Video Optimizer still communicates via SSL to the web server. The communication is SSL (encrypted) from web browser to Video Optimizer and SSL (encrypted) from Video Optimizer to the web server.

**Connection Set-up for Secure Trace:**

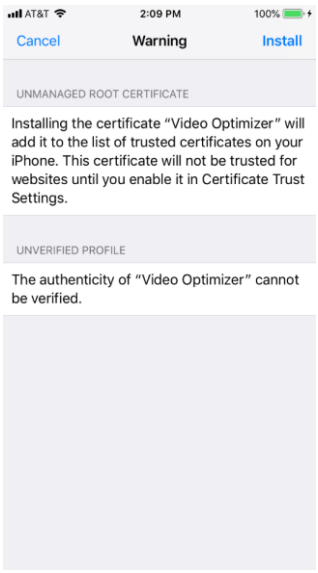
**Pre-Requisite**- MITM Proxy setup- Refer to section **(3.2.3.2)**

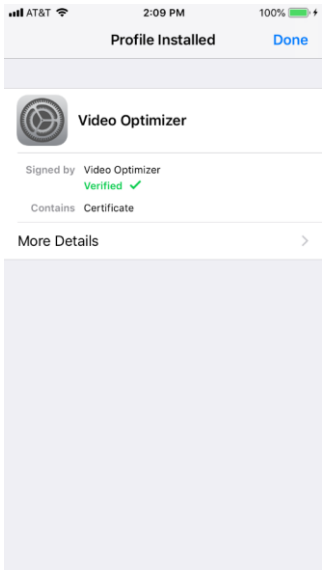
→Navigate to Menu>Start collector to launch the start collector window.

→On the start collector window Secure(yes) should display MITM status with Internet Sharing Status, Server address IP and server port and then enter a valid trace name and then click start. Refer to the screenshot below.

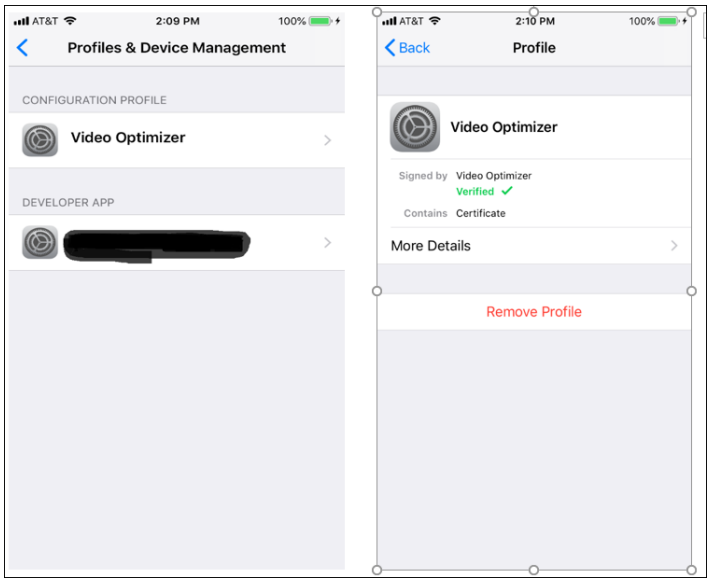



→Navigate to Safari and type in IP address 192.168.2.1:9091 to install the SSL certificate on the phone.

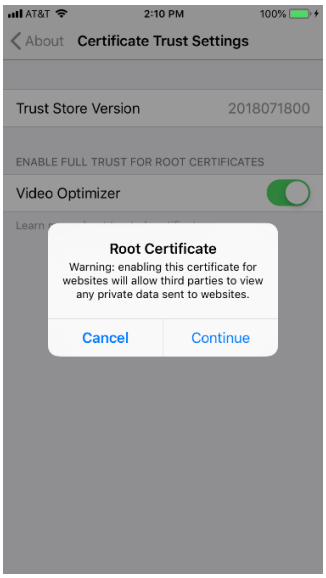



→Navigate to settings>general>Profiles and Device Management to verify that Video Optimizer is present in the configuration profile and click on it to see that the certificate is verified.

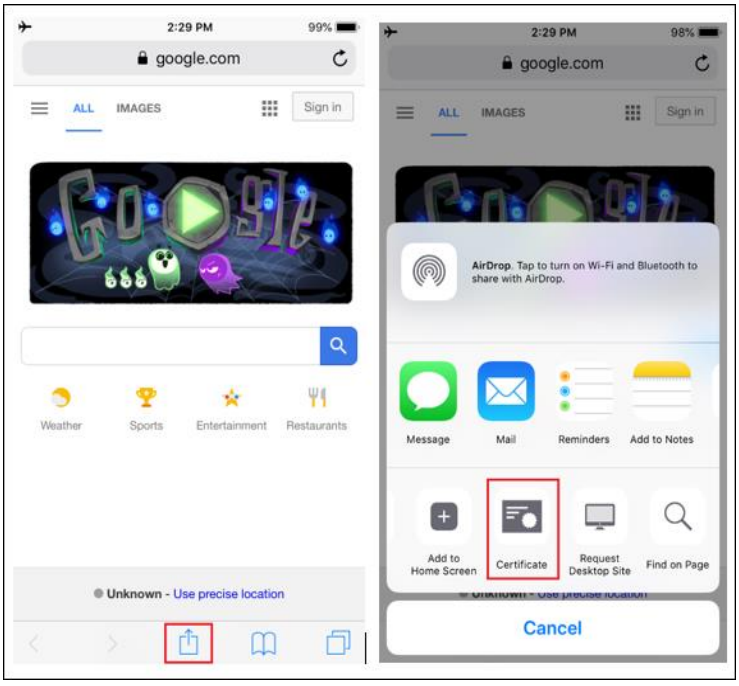



→ If you are on iOS 10.3 or higher, open the Settings>General > About > Certificate Trust Settings, and find the Video Optimizer Proxy certificate, switch it "ON" to enable full trust for it.

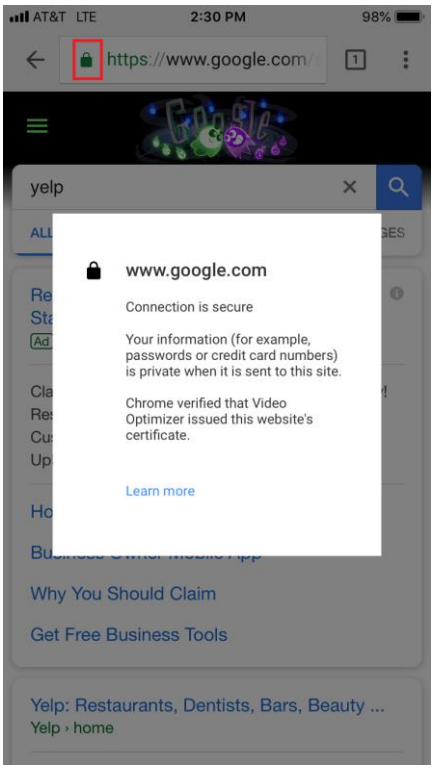



→Open Safari or Google to validate the certificate.

→On Safari following screenshots display the method to validate secured




→On Google click on the lock symbol to validate that the connection is secure.




## 3.3. Testing your Application

Now that the Video Optimizer is running on your mobile device, you can run tests on your mobile application or website.

When testing your application, use your app like a typical user. Test logging in, or other common use cases. Follow general test cases that you may run on each build to ensure that the application functions properly.

***Note:*** All the data transmitted during your test is being recorded, and the screen may also be recorded if you have chosen to record video. Please only use test data while using the Video Optimizer, or your private credentials will become available to anyone who has access to the trace.

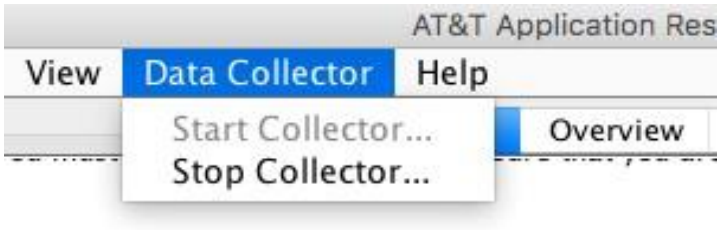
## 3.4. Ending the Trace

When you have completed the Video Optimizer recording, use the following steps to stop the trace, confirm that the trace files have been pulled to your computer, and begin your Video Optimizer analysis.

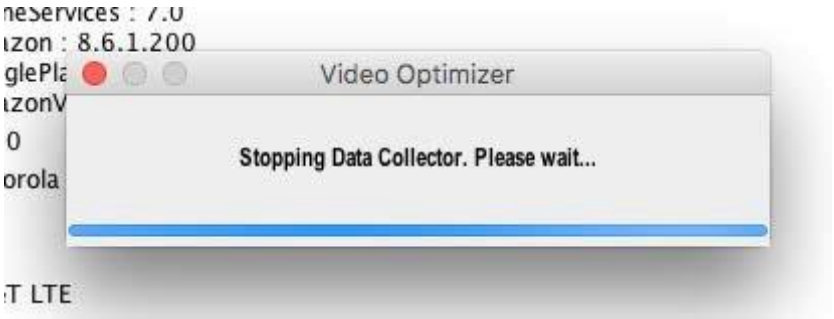
Step 1. Stop the trace in one of the following ways.

a. If you are collecting video: Click the stop button on the video viewer (as shown in **Figure**).

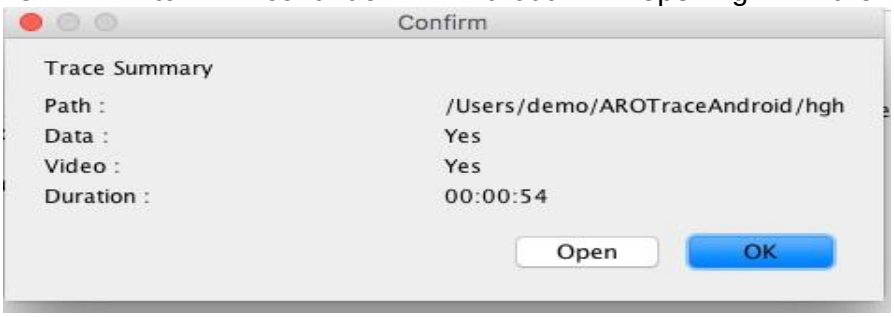

b. If you elected to *not* collect video: Stop the trace using the Stop Collector option on the Data Collector menu (as shown in **Figure**).




Step 2. When the trace collection is stopped, the files will be pulled across to your computer. Video Optimizer will create a folder in your root directory called VOTrace<osName> and will save the trace there.



Step 3. Once the files have been pulled across (this may take a while if you recorded HD or SD video due to the large file size), the Trace Summary is displayed **Figure**), and you can open the trace in the Video Optimizer analyzer or click "OK" to continue without opening the trace




# 4. Analyzing a Trace

Video Optimizer will perform an analysis of your network trace, including the radio resource and energy usage of the applications run during the trace. Video Optimizer works from application traces gathered through its Data Collector to do the analysis.

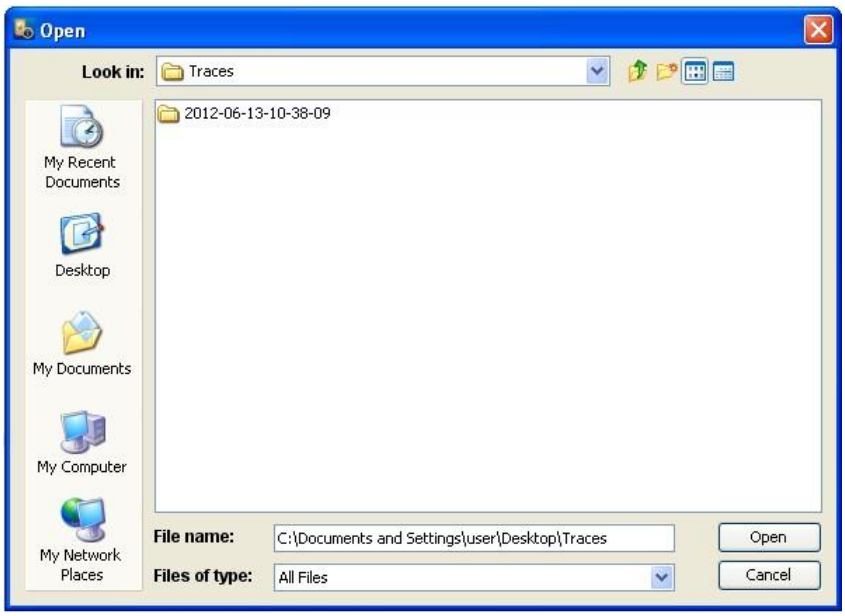
Video Optimizer provides the following:

- Visibility into radio resource and energy utilization.
- Benchmarking of resource efficiencies.
- Automatic diagnosis of application inefficiencies.

## 4.1. Opening a Trace File

When you open a trace file in Video Optimizer, the data is evaluated against a set of recommended best practices. Video Optimizer looks at how your application (and your server) is handling caching, how you are managing the network connections for your application, how your app is using HTML, whether your app is treating data securely, and how it is handling video streaming. Use the following procedure to open a trace file.

Step 1. Select Open Trace from the File menu to display the Open dialog box.

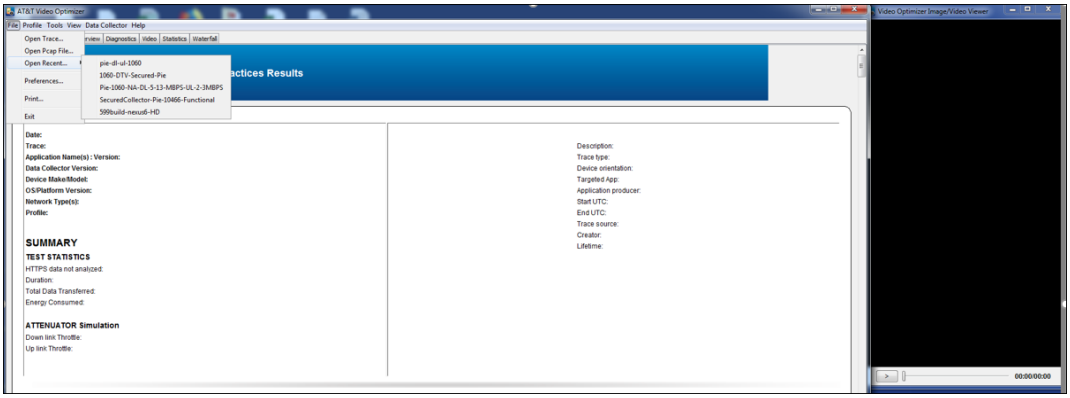



Step 2. Select a trace folder and click Open. The trace files are loaded, and Video Optimizer begins analyzing the data.

**Note:** The time required for Video Optimizer to complete the analysis depends on the size of the trace. As soon as the analysis is complete, all the content tabs in the Video Optimizer are updated with the analysis results.

Step 3. *Optional*. Select Open pcap File from the File menu (instead of Open Trace) to open pcap trace files and trace files that have been collected using the Microsoft Network Monitor. These files allow you to view statistical and analytical data for the trace based on the packet information in the file, but you will not be able to view any video information in the Viewer and you will not see any information for peripheral applications (i.e. GPS or Bluetooth). **Note**: Out of Memory (OOM) or "Trace too big to load" message can occur for large trace loading. When these errors occur, try closing Video Optimizer and reloading the trace. Also consider collecting multiple, smaller, more targeted traces

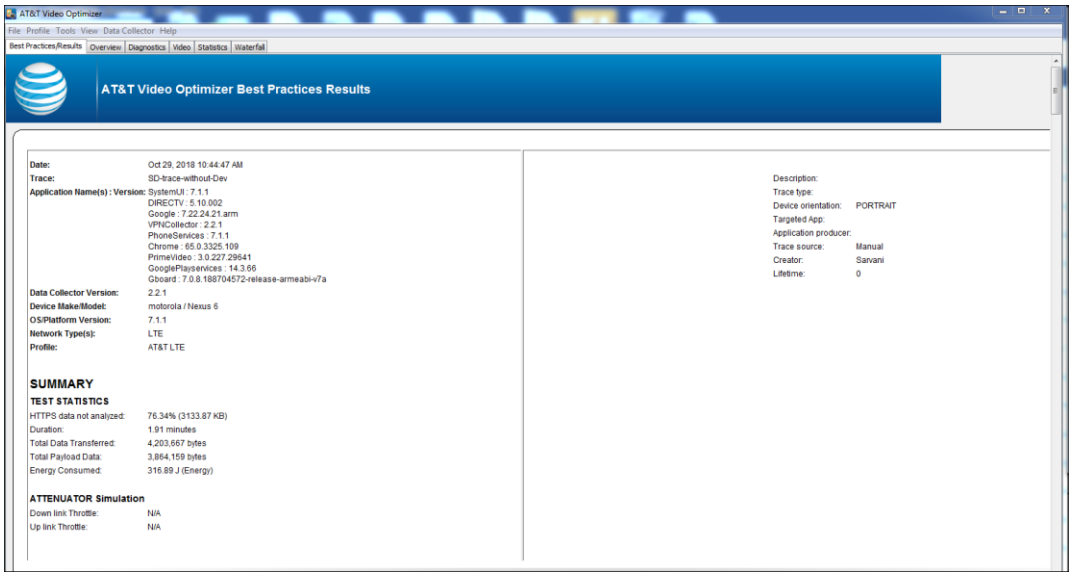
Step 4. *Open Recent file*: User has the option to open the recent 5 traces from Analyzer which allows user to find the last few traces.



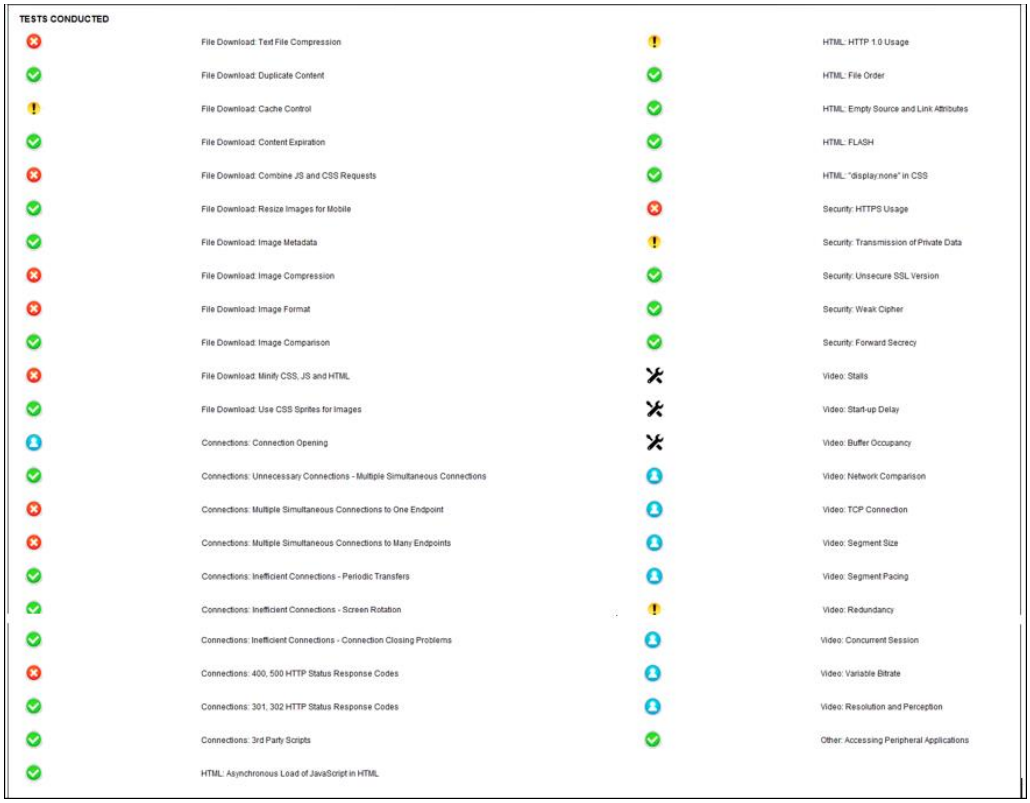
Open Recent file: When user re-names any file which was opened and displayed in the open recent file, the file name will be removed from list. The file name will not appear in the open recent until the user re-loads the trace. Also, user can re-name the trace with any special characters and load it in the Analyzer.

## 4.2. Best Practices Results

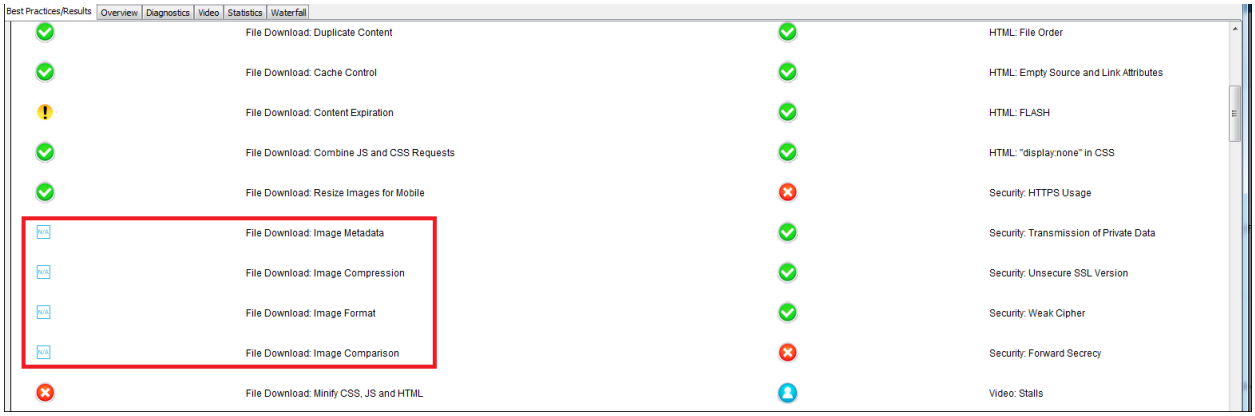
Video Optimizer trace analysis starts at a high level, with the Best Practices Results, and then it helps you to pinpoint the issue as much as is required to solve the problem. The high-level views work well for executive readouts, while moderate digging might be useful for a deeper dive, and for a deep technical understanding – you can dig even deeper.



The top of the Best Practices Results tab tells you some basic information about the trace such as the date, file name, and information on the device that was used for the test. The Summary section reveals a few high-level statistics recorded in the trace. Right below this is the Tests Conducted section, which contains a summary of the best practices so that you can quickly determine which best practices failed. The below screenshot refers to a video trace that is uploaded in the analyzer.

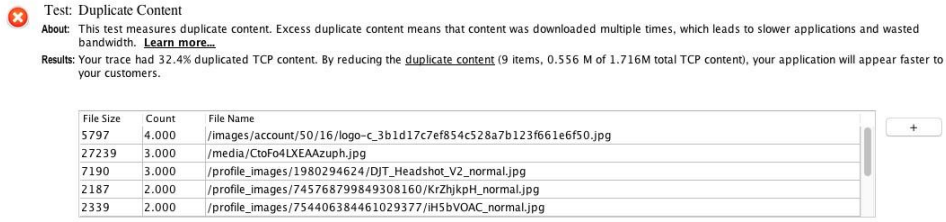



A 'N/A' icon is displayed to indicate that a Best Practice is not applicable when a trace does not have relevant data available for analysis. Refer the image below for reference



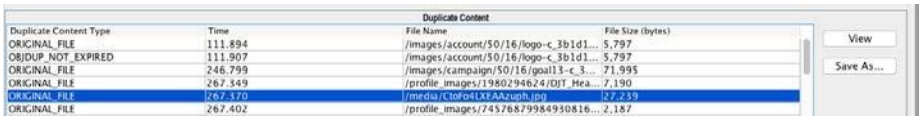
Generally, the Tests Conducted section of the Best Practices tab is the best place to start to improve your application.

Step 1. Click on the text for the Best Practices test to go to a larger summary. A green checkmark means that the test was successful, and a red x means that the test failed. For example, here is a trace that failed the Duplicate Content best practice.

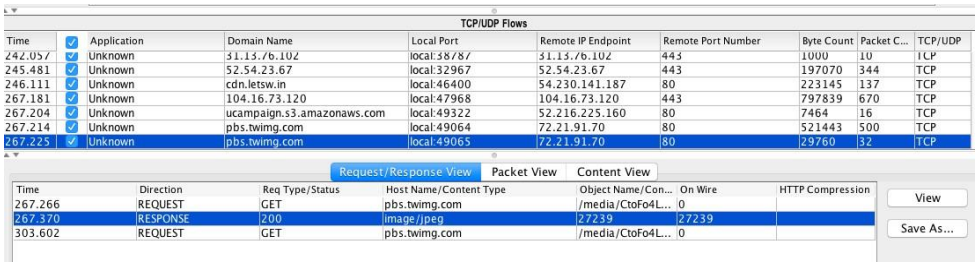



Step 2. Review the About section to learn more about this best practice. Review the Results section for details about how to fix the problem. This example has a table listing files that were sent multiple times including the count of downloads and the file name.

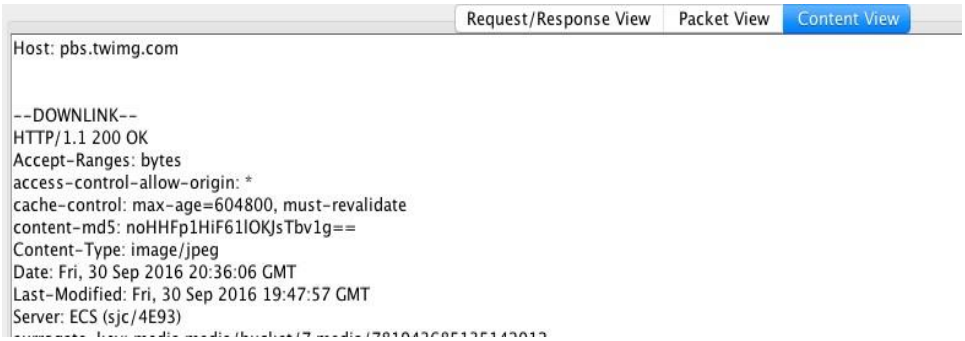
Step 3. Click one of the table entries for more information—every best practice table will let you do this. In this example, you are taken to the full list of duplicate content in the Overview tab, from which you may be able to determine the time ranges or other details to figure out the issue.




Step 4. Select a row in the table and click View to view the contents of the file. This helps when a file name is obscure, and you don't really know what the file is or the type of content. The Diagnostics tab is displayed, and you can see the request/response for the file.



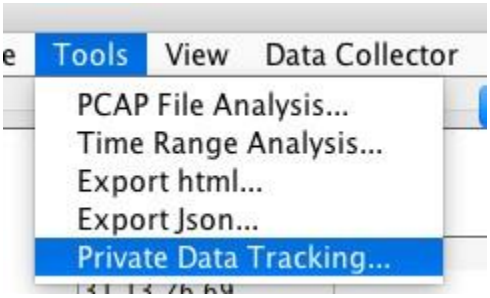
Step 5. Click Content View to see the headers. In **Figure**, the top table shows the IP and domain (pbs.twimg.com) indicating a request from Twitter's image servers. The bottom table shows the request/response. In this example, the file is coming from Twitter. Check the cache-control parameter in the DOWNLINK header to see if the file has the correct cache headers. In **Figure**, cache-control shows a max-age of 604800s—a cache timeframe of one week—so in this application, the image cache may not be properly set to store files locally on the phone.



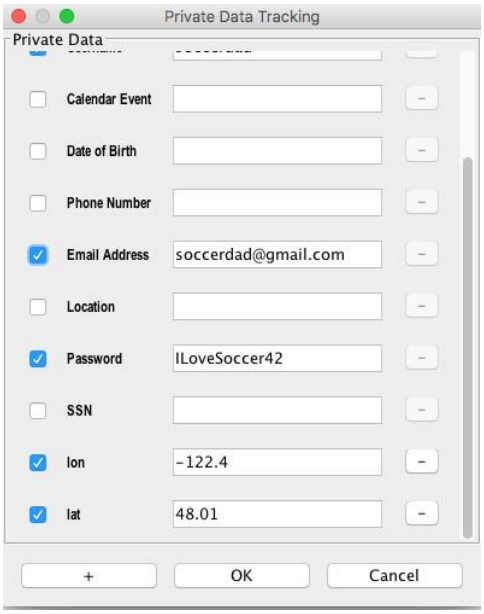

## 4.3. Private Data Tracking

The Transmission of Private Data best practice test allows you to examine all the decrypted and clear text data files for leakage of private data.

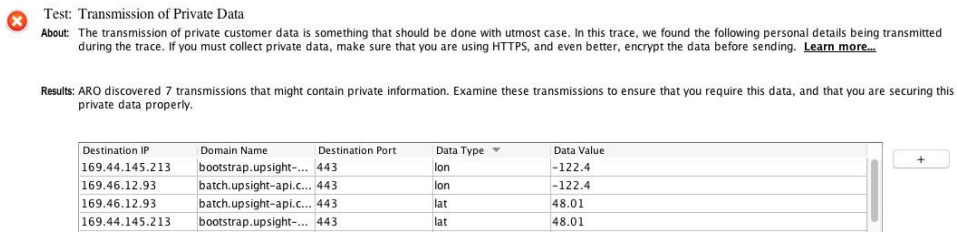
Step 1. Select Private Data Tracking from the Tools menu.



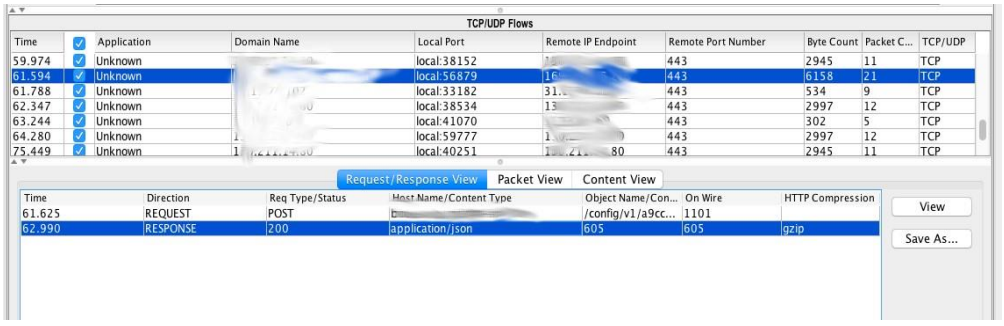

Step 2. In the Private Data Tracking dialog, complete the fields that apply to your trace.



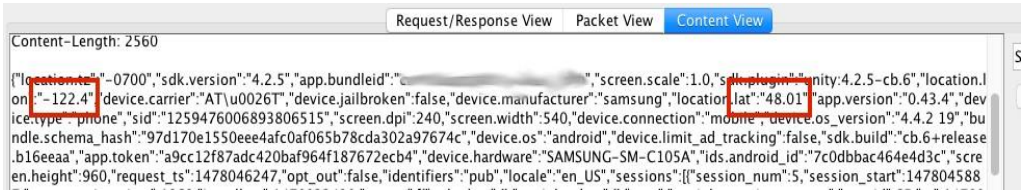

Step 3. Click OK to analyze the trace for these values. A report will be provided for you in the best practices tab. The Data Types and Data Values in **Figure** were found in the Video Optimizer trace because they were entered in the Private Data Tracking dialog.




Step 4. Click one of the rows in the table to see the Diagnostics tab. The top table shows the TCP flow and the bottom table shows the Request/Response View.



Step 5. Click Content View to see the POST data with the private data transmission in the bottom table.




## 4.4. Video Analysis

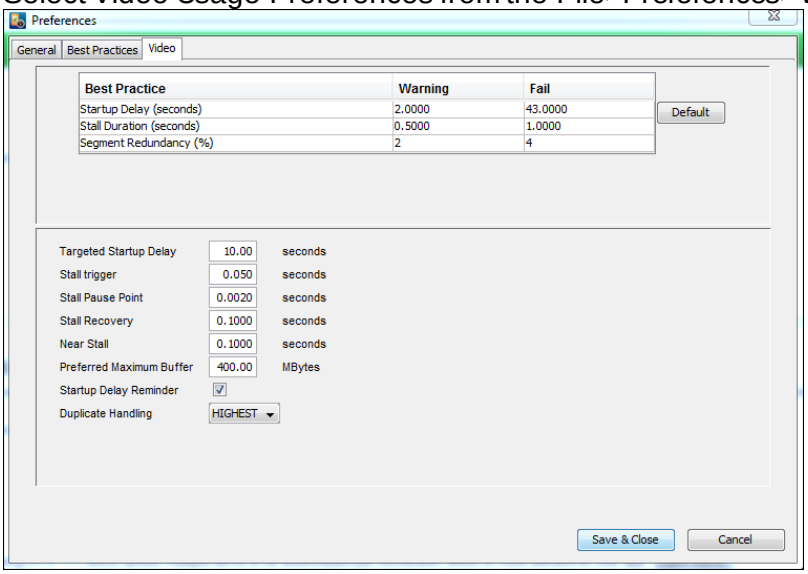
Video Optimizer contains several functions to analyze the delivery of streaming video in your mobile application. These specialized tools have some unique features that require additional configuration.

When testing video, your trace should ideally be of a DRM free video, and the application should allow screen recording. Video Optimizer can perform several analyses without these two features, but the examples in this section are of a trace with a DRM free video, and the recorded screen.

### 4.4.1. Video Analysis Setup

Begin by opening a trace with video content in it, but before beginning analysis, there are a few manual steps required.

Step 1. Select Video Usage Preferences from the File>Preferences>Video.




Step 2. Enter the **Startup Delay** in seconds. This is the KPI (Key Performance Indicator) requirement for how quickly a video should begin playing. The default is set at 10 seconds. A video that takes longer to begin playing

than the value set for Startup Delay will fail the Video Startup best practice.

Step 3. Enter the **Stall trigger** in seconds. The default value is 0.05s (50ms). Stalls that are under the threshold value set for Stall trigger will not be counted in the best practice. For example, the value in **Figure** means that stalls under 0.05s will not be counted in the best practice.

Step 4. Enter the **Preferred Maximum Buffer**. Video Optimizer measures the amount of video stored in the buffer. Here you can set the max buffer size (in MB). For example, the value in **Figure** denotes a 400 MB video buffer.

Step 5. Click OK.

### 4.4.2. Configuration Required

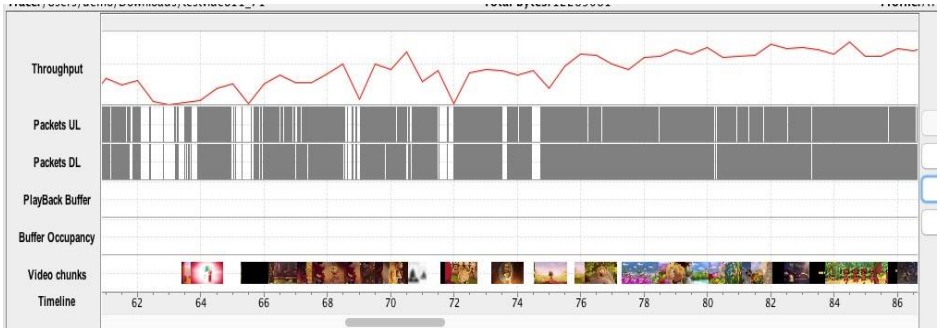
From the best practices tab user can set the start-up delay by clicking on the icon of any these three best practices Video: Stalls, Start-up Delay, Buffer Occupancy. Refer to the screenshot below


### 4.4.3. Analyzing a Video

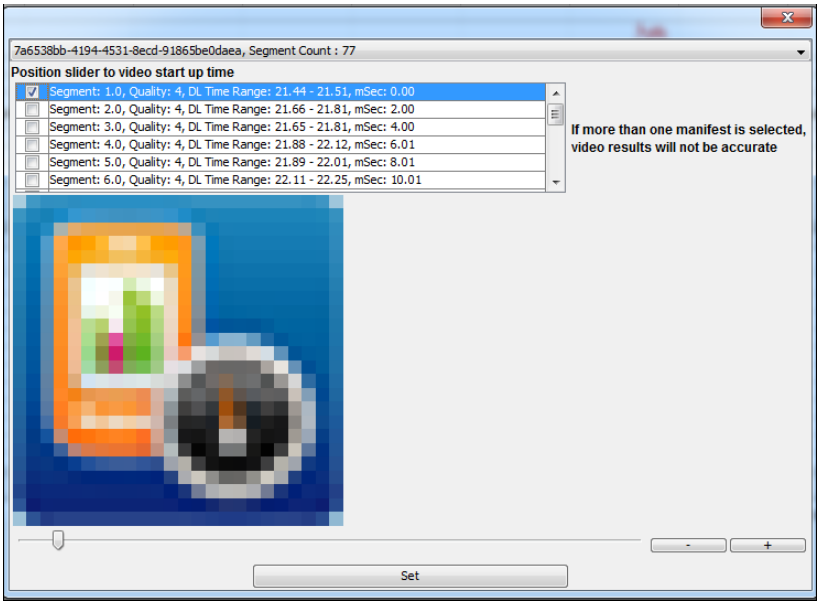
Switch to the Diagnostics tab to further analyze your video.

Step 1. Select Options from the View menu.

Step 2. Select Video View from the dialog. This will add six lines to the diagnostics chart that are designed specifically to look at video.

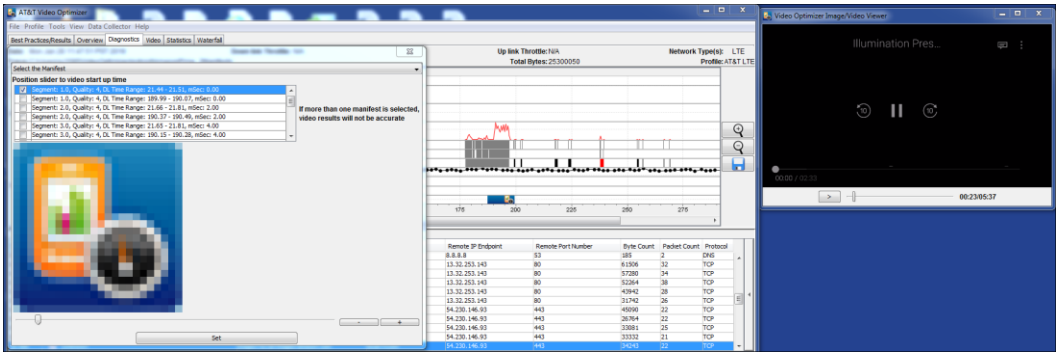



Step 3. Click the first segment in the Video segments (bottom row). If the video you tested was DRM free, the view will be of the first frame in each chunk.




Step 4. In the above image the segment started to download at 21.44ms

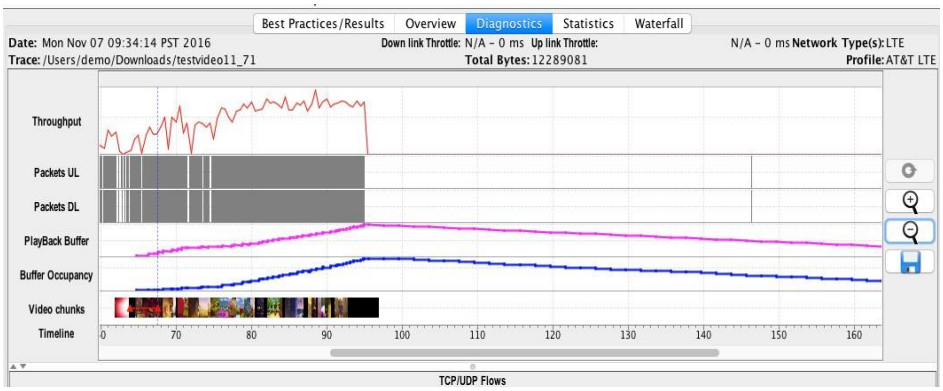
Step 5. Move the slider to see the video viewer move through the captured video. Match the first frame of segment 1 with the screen recording and press Set. This allows Video Optimizer to do some initial calculations on the size of the video buffer before the video starts to play.



**4.4.3.1.** The second and third rows from the bottom of the Diagnostics tab will now be populated with data. Zoom out for a better view.

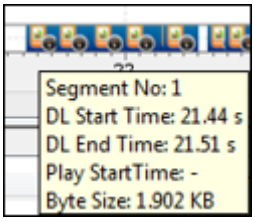
### 4.4.3.2. Viewing Buffer Details

The Diagnostics tab shows the Playback Buffer (the seconds of video stored locally) and the Buffer Occupancy (the KB of video stored locally). Examine the Video chunks line closely—in **Figure** there is a red line starting at chunk 1 and ending when the chunk appears on the screen—in this case, around 68s. This allows Video Optimizer to calculate if the video buffer runs out of video time or KB. When this occurs, there is no more video to play, and a stall occurs to the viewer. Video Optimizer will use the startup occupancy found at the start to estimate when the video will resume playing.

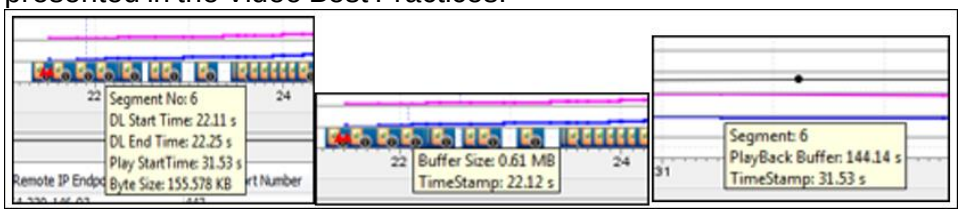



### *4.4.3.3. Getting Video Information*

1. Move the pointer over a video chunk to determine when the download started, stopped, played on the screen, and the number of KB in the chunk.

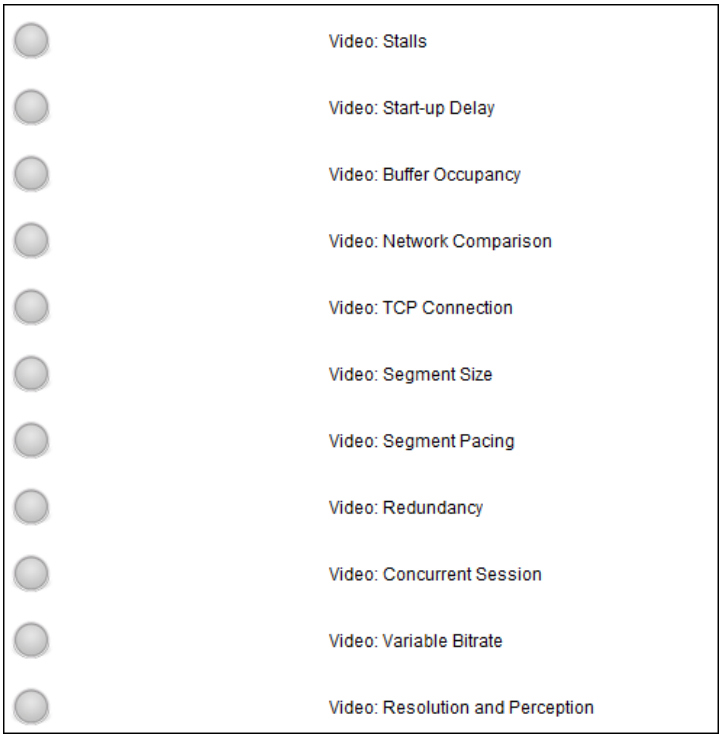


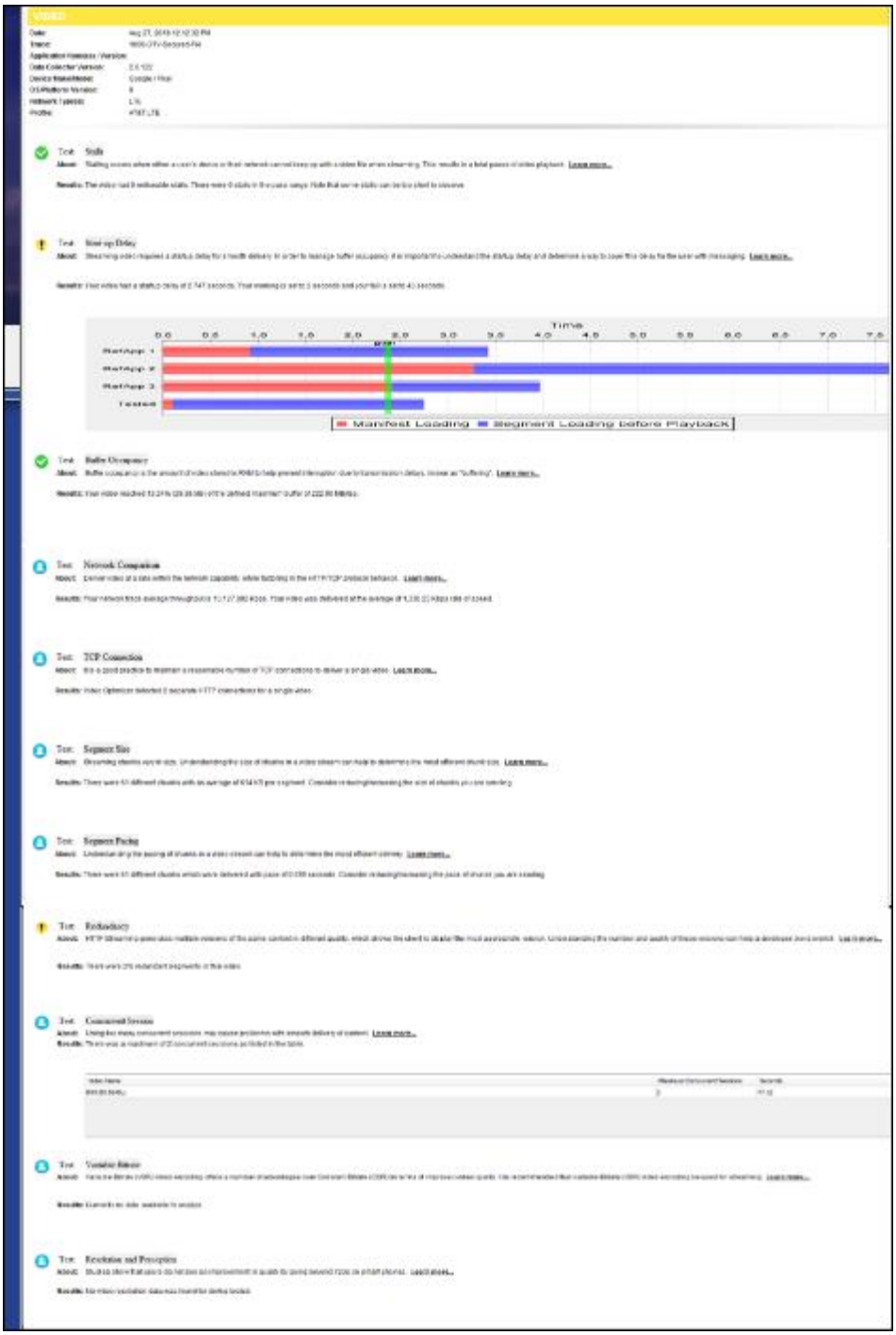
2. Point to the Play Back Buffer and Buffer Occupancy lines for more information about how much playback is stored in the buffer and how much is used. In the Image below for segment 6, at 22.12s, the video buffer has 0.61MB of data, which contains 144.14s of video playback time. These steps will also allow Video Optimizer to calculate the data presented in the Video Best Practices.




### 4.4.4. Video Best Practices

The following are examples of the best practices in Video Optimizer. These best practices are still very experimental and are simply providing information. As Video Optimizer matures, these will become fully fledged tests with pass and fail criteria.

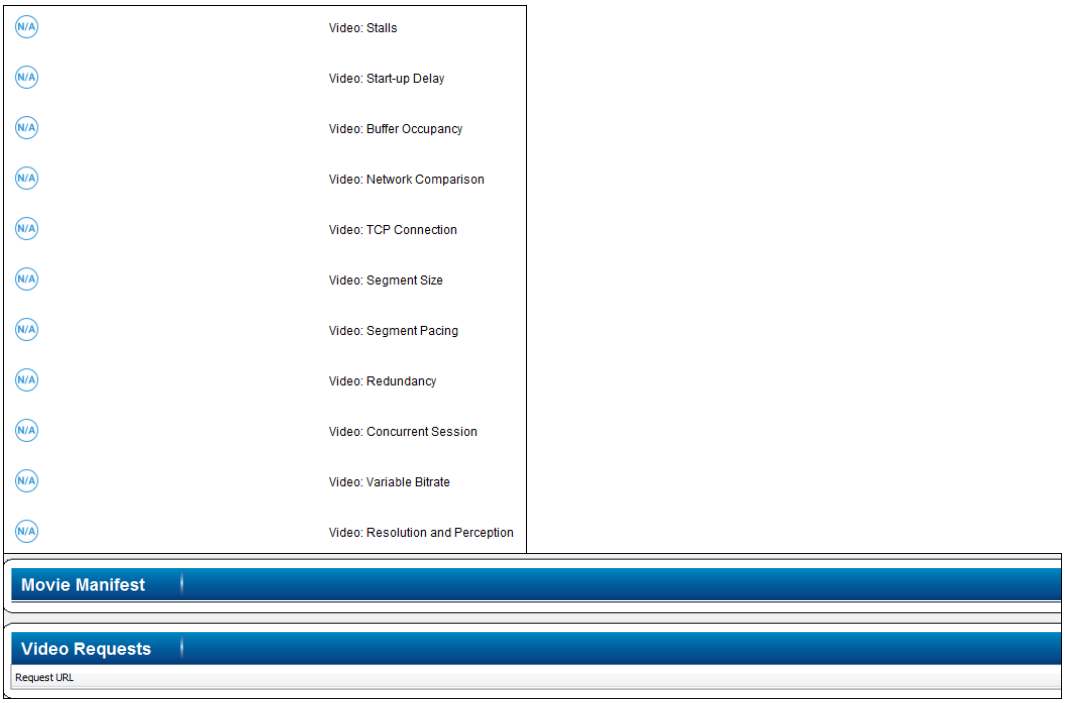





#### 4.4.4.1. *Video Conditions*

1)There is no video in the trace

- [NA] across all VBP
- Response: No streaming video data found.
- no manifest
- no segments
- no urls




2)Video not handled in the trace
- [Config_Required] across all VBP (no analysis)
- Response:Video Optimizer was unable to analyze video manifest file(s) Hint: look for ways to locate manifest/segment information with the Video Parser Wizard. Click here to select Request URL <a href="selectManifest"> Manifest</a> on the Video tab
    - no manifest
    - no segments
    - urls exist

3)There is only one valid manifest created.
- [Config_Required] across all VBP (no analysis)
- Response: Invalid manifests. Video Optimizer did not have enough information for analyzing streaming video. Hint: look for ways to locate segment information with the Video Parser Wizard. Click here to select Request URL <a href="selectManifest"> Manifest</a> on the Video tab.
- manifest - invalid
- segments
- urls exist

4)There is only one valid video manifest created
- apply tests to all VBP for selected manifest only
- manifest-Valid
- segments
- urls Exist

5)There is only one valid manifest but de-selected video manifests
- [Config_Required] across all VBP (no analysis)

- Response: No manifest is selected. Please select a <a href="selectManifest"> Manifest</a> on the Video tab.
- manifest
- segments
- urls exist

6) There are Multiple valid manifests
   **All selected**
   - [Config_Required] across all VBP (no analysis)
   - Response: Please select only one manifest on the Video tab. Click here to select a <a href="selectManifest"> Manifest</a> on the Video tab.
   - none selected
   - [Config_Required] across all VBP (no analysis)
   - No manifest is selected. Please select a <a href="selectManifest"> Manifest</a> on the Video tab.

7) There are Invalid with valid manifest
   - one Selected manifest
     - apply tests to all VBP for selected manifest only
     - ignore invalid manifests

8) There are invalid with valid manifest
   - none Selected manifest
   - [Config_Required] across all VBP (no analysis)
   - Response: No valid manifest has been selected. Please select a <a href="selectManifest"> Manifest</a> on the Video tab. Note: There are invalid manifests detected. Video Optimizer did not have enough information for analyzing streaming video. Hint: look for ways to locate segment information with the Video Parser Wizard. Click here to select Request URL <a href="selectManifest"> Manifest</a> on the Video tab.

# 5. Video Optimizer Reference

The following sections are a complete reference of the Menus and Content Tabs that are available in Video Optimizer.

## 5.1. Menu

Video Optimizer has the following menus.

| Menu | Description |
|---|---|
| File | Contains options for opening trace, opening Pcap File, Preferences, Open Recent, Printing results, and Exiting the application. |
| Profile | Contains options for loading and customizing device/trace profiles. |

| | |
|---|---|
| Tools | Contains options for running a Pcap file analysis, running a Time Range Analysis, exporting analytical data in the JSON format. |
| View | Contains options for displaying the video viewer, filtering the set of data that appears in the analysis and diagnostics based on application, IP address, and time range, and configuring which data should appear in the Diagnostics Chart. |
| Data Collector | Contains options for accessing the Video Optimizer Data Collector. |
| Help | Contains options for displaying the Video Optimizer version, the FAQ page, User Documentation, Dependencies, Forum, Support, Downloads, and other Help documentation. |

### 5.1.1. File Menu

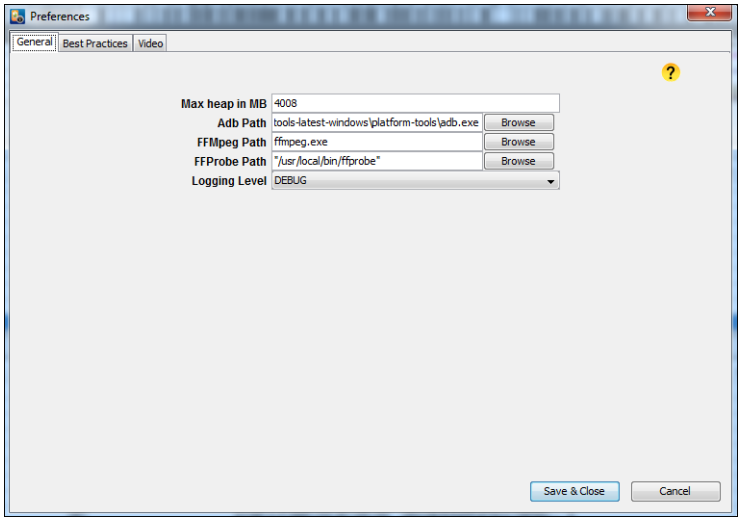
File menu contains the following selections.

| Selection | Description |
|---|---|
| | |
| Open Trace | Opens a file for trace analysis. When this menu item is selected, a dialog box is displayed that prompts you to select the location of the trace folder containing the trace files. You must select a trace file using this command before you can view the trace analysis of that file. |
| Open Pcap File | Opens a Pcap file for packet data analysis and also opens files that have been collected using the Windows Network Monitor (NetMon). When this menu item is selected, a dialog box is displayed that prompts you to select the location of the Pcap file. You must select a Pcap file using this command before you can view a packet data analysis of that file.<br><br>**Note:** To open a file that was collected using NetMon, you must have NetMon installed on your machine. |
| Open Recent | User has the option to open the recent 5 traces from Analyzer which allows user to find the last few traces. |
| Preferences | Opens a Preferences dialog where you can choose the Best Practices that you want to analyze and set a path for the Android Debug Bridge (ADB). |

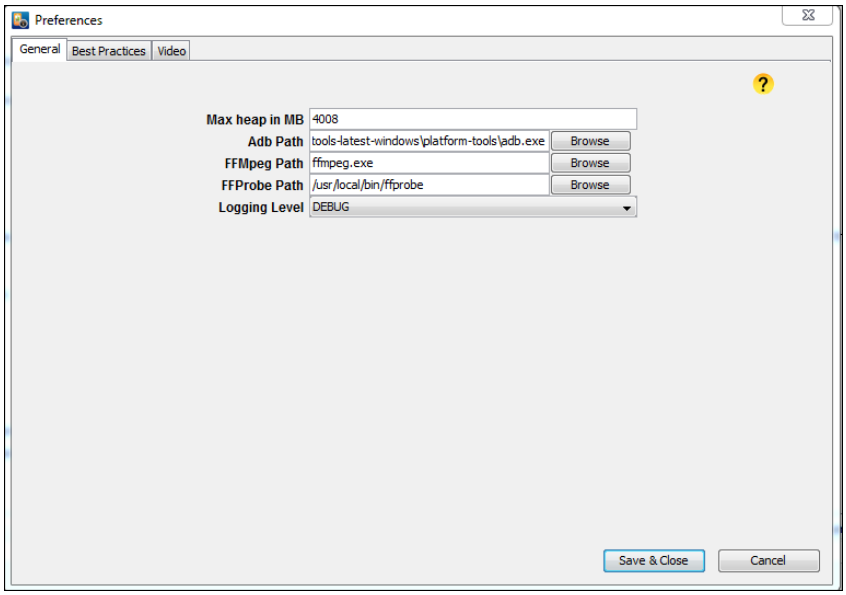
| | |
|---|---|
| Print | Prints the results that are displayed when the Best Practices or Statistics tab is selected. This menu option is only enabled when the Best Practices or Statistics tab is selected. |
| Exit | Exits the Video Optimizer application. |

### 5.1.2. Preferences

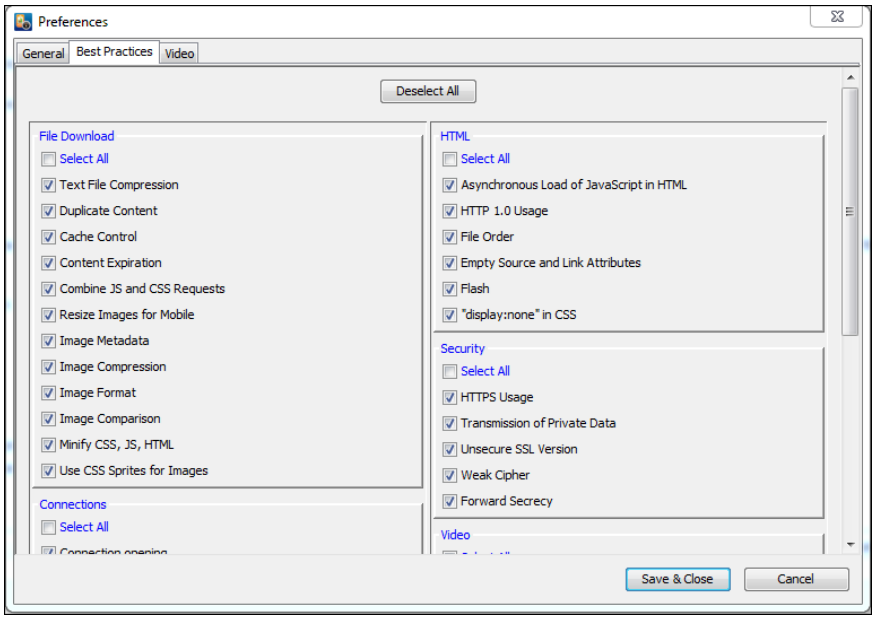
This feature opens a dialog with three tabs available, General, Best Practices, and Video. Select Preferences from the File menu. Select the General tab.



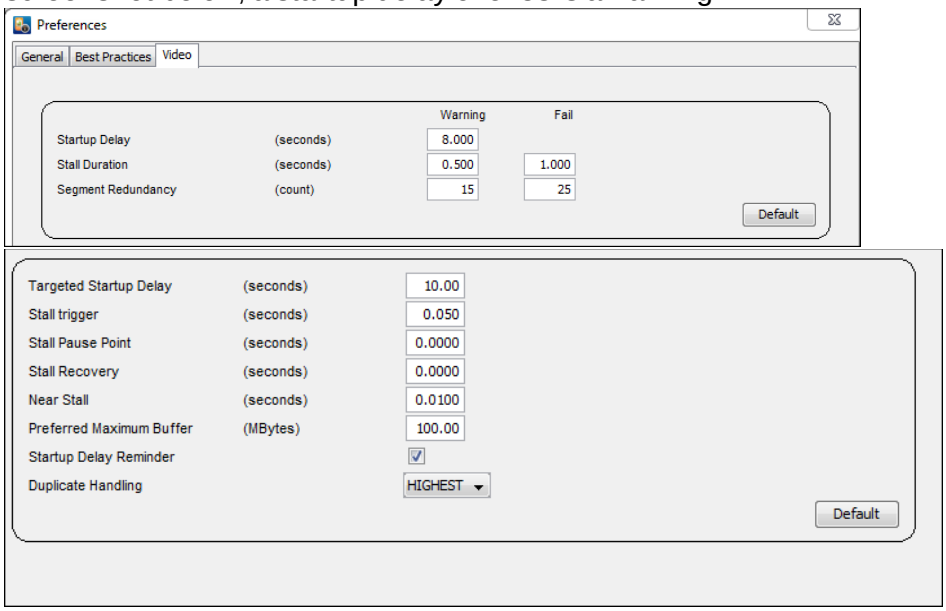

- Enter the **Max heap in MB**. Here you can set the maximum memory heap size for Video Optimizer (in MB). For example, the value in **Figure above** denotes 4008 as the heap size.
- Enter the **Adb Path**. This is the location of the Android Debug Bridge (ADB) on your computer.
- Note: When entering the ADB path in this dialog, you must include the “adb.exe” at the end of the path.




- Dumpcap path: required on iOS for trace evaluation.
- iDeviceScreenshot Path: required on iOS for screen capture.
- ffmpeg path: required for video analysis
- ffprobe path required for video analysis
- iOS certificate: required for iOS HD video capture (see set up for more details)
- Select the Best Practices tab.
- Choose the best practices that you want to analyze while opening the trace.

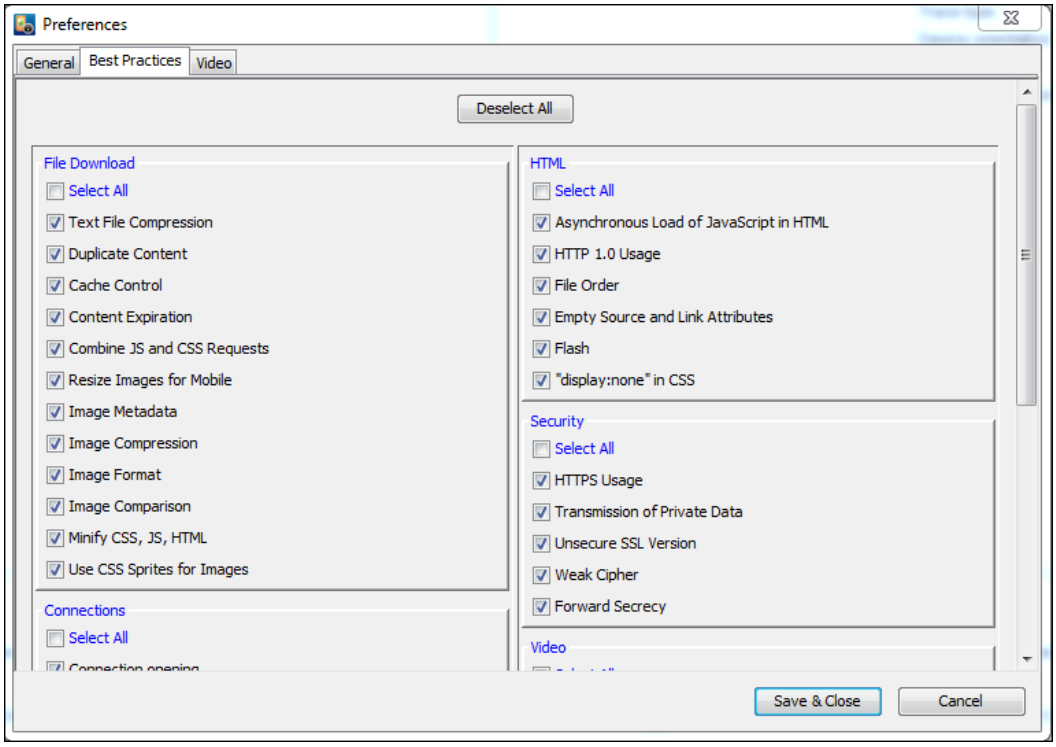


- Select Video
- Here you can modify the parameters for pass/fail criteria for some of the video best practices. In the screenshot below, a startup delay over 8s is a warning.




- When you save the changes in the Preferences dialog, a refresh is done on the trace and the selected values are persisted across Video Optimizer launches. This means that only the selected best practices are run while opening the trace. Also, we limit the video segments to be displayed in the diagnostics and Video tabs when any of the Video best practices or any of the Image best practices such as Compression, Metadata or Format are selected for analysis.

(i) Select/Deselect All
User has the ability to select or de-select all the best practices preferences.




### 5.1.3. Profile Menu

The Profile menu contains the following selections.
Note: Profiles are intended for advanced users of Video Optimizer trace analysis.

| Selection | Description |
|---|---|
| Load | Loads the selected Profile. |
| Customize | Displays the attribute values of the selected Profile and allows you to edit them. |

i. AT&T 3G Profile Network and Device Attributes
ii. The following network attributes are defined for the AT&T 3G profile.

| Network Attribute | Description |
|---|---|
| Carrier | The network carrier for the device. |

| | |
|---|---|
| DCH (Active)->FACH (Standby) timer (sec) | The amount of time (in seconds) used when the RRC state changes from (direct channel) DCH to (forward access channel) FACH. |
| FACH (Standby)->IDLE timer (sec) | The amount of time (in seconds) used when the RRC state changes from FACH (Forward access channel) to IDLE. |
| Min IDLE->DCH (Active) promotion delay (sec) | The minimum amount of time used (in seconds) when the RRC state is promoted from IDLE to DCH (Active). |
| Avg IDLE->DCH (Active) promotion delay (sec) | The average amount of time used (in seconds) when the RRC state is promoted from IDLE to DCH (Active). |
| Max IDLE->DCH (Active) promotion delay (sec) | The maximum amount of time used (in seconds) when the RRC state is promoted from IDLE to DCH (Active). |
| Min FACH (Standby)->DCH (Active) promotion delay (sec) | The minimum amount of time used (in seconds) when the RRC state is promoted from IDLE to DCH (Active) and FACH (Forward access channel) to DCH (Active). |
| Avg FACH (Standby)->DCH (Active) promotion delay (sec) | The average amount of time used (in seconds) when the RRC state is promoted from IDLE to DCH (Active) and FACH (Forward access channel) to DCH (Active). |
| Max FACH (Standby)->DCH (Active) promotion delay (sec) | The maximum amount of time used (in seconds) when the RRC state is promoted from IDLE to DCH (Active) and FACH (Forward access channel) to DCH (Active). |
| RLC threshold for uplink (bytes) | The RLC threshold value (in bytes) for uplink. |
| RLC threshold for downlink (bytes) | The RLC threshold value (in bytes) for downlink. |
| Threshold for resetting DCH (Active) timer (bytes) | The threshold for resetting the DCH (Active) timer (in bytes). |
| Timing window for resetting DCH (Active) timer (sec) | The timing window for resetting the DCH (Active) timer (in seconds). |
| **Network Attribute** | **Description** |
| RLC consumption rate (^2) for uplink | The RLC consumption rate (^2) for uplink. |
| RLC consumption rate (^1) for uplink | The RLC consumption rate (^1) for uplink. |

| | |
|---|---|
| RLC consumption rate (^0) for uplink | The RLC consumption rate (^0) for uplink. |
| RLC consumption rate (^2) for downlink | The RLC consumption rate (^2) for downlink. |
| RLC consumption rate (^1) for downlink | The RLC consumption rate (^1) for downlink. |
| RLC consumption rate (^0) for downlink | The RLC consumption rate (^0) for downlink. |
| Time delta for throughput calculations (sec) | The time delta (in seconds) used for calculating throughput. |
| Threshold for defining a burst (sec) | The time threshold (in seconds) used for defining a burst. |
| Threshold for defining a long burst (sec) | The time threshold (in seconds) used for defining a long burst. |
| Threshold for user input window (sec) | The time threshold (in seconds) used for calculating user input. |
| Periodical Transfer Analysis - Min size of periodical clusters (sec) | The minimum size (in seconds) of a cluster of periodical transfers. |
| Periodical Transfer Analysis - Max tolerable variation for periodical transfers (sec) | The maximum tolerable variation (in seconds) used for calculating periodical transfers. |
| Periodical Transfer Analysis - Min number of periodical transfers | The minimum number of periodical transfers. |
| Threshold for duration of a large burst (sec) | The duration threshold (in seconds) used for defining a large burst. |
| Threshold for size of a large burst (bytes) | The size threshold (in bytes) used for defining a large burst. |
| Threshold for close spaced bursts (sec) | The threshold (in seconds) used for defining a close spaced burst. |

The following device attributes are defined for the AT&T 3G profile.

| **Device Attribute** | **Description** |
|---|---|
| Device Name | The make and model of the device. |
| **Device Attribute** | **Description** |
| DCH (Active) Power (w) | The amount of power (in watts) that should be used when the RRC state is DCH (Active). |

| | |
|---|---|
| FACH (Standby) Power (w) | The amount of power (in watts) that should be used when the RRC state is FACH (Standby). |
| IDLE Power (w) | The amount of power (in watts) that should be used when the RRC state is IDLE. |
| Average power for IDLE->DCH (Active) promotion (w) | The average amount of energy used when the RRC state is promoted from IDLE to DCH (Active). |
| Average power for FACH (Standby)>DCH (Active) promotion (w) | The average amount of power (in watts) that should be used when the RRC state is promoted from FACH (Standby) to DCH (Active). |
| Average power for active GPS (w) | The average amount of power (in watts) for active GPS. |
| Average power for standby GPS (w) | The average amount of power (in watts) for standby GPS. |
| Average power when camera is on (w) | The average amount of power (in watts) when the camera is on. |
| Average power for active Bluetooth (w) | The average amount of power (in watts) for active Bluetooth. |
| Average power for standby Bluetooth (w) | The average amount of power (in watts) for standby Bluetooth. |
| Average power when screen is on (w) | The average amount of power (in watts) when the screen is on. |

iii. AT&T LTE Profile Network and Device Attributes

iv. The following network attributes are defined for the AT&T LTE profile.

| Network Attribute | Description |
|---|---|

| | |
|---|---|
| Carrier | The network carrier for the device. |
| Promotion time from Idle to CR (sec) | The amount of time (in seconds) spent in promotion from the IDLE state to the CR state. |
| Time of inactivity from CR before DRX (sec) | The amount of inactive time (in seconds) spent in the CR state before changing to the DRX state. |
| Time in short DRX (sec) | The amount of time (in seconds) spent in the Short DRX state. |
| Ping length during DRX (sec) | The length of a ping (in seconds) during the DRX state. |
| Time in Long DRX (sec) | The amount of time (in seconds) spent in the Long DRX state. |
| Ping length in IDLE (sec) | The length of a ping (in seconds) during the IDLE state. |
| Period between pings DRX Short (sec) | The length of the period between pings (in seconds) in the Short DRX state. |
| Period between pings DRX Long (sec) | The length of the period between pings (in seconds) in the Long DRX state. |
| Period between pings IDLE (sec) | The length of the period between pings (in seconds) in the IDLE state. |
| Time delta for throughput calculations (sec) | The time delta (in seconds) used for calculating throughput. |
| Threshold for defining a burst (sec) | The time threshold (in seconds) used for defining a burst. |
| Threshold for defining a long burst (sec) | The time threshold (in seconds) used for defining a long burst. |
| **Network Attribute** | **Description** |
| Threshold for user input window (sec) | The time threshold (in seconds) used for calculating user input. |

| | |
|---|---|
| Min cycle for periodical transfers (sec) | The minimum cycle time (in seconds) used for calculating periodical transfers. |
| Max tolerable variation for periodical transfers (sec) | The maximum tolerable variation (in seconds) used for calculating periodical transfers. |
| Min observed samples for periodical transfers | The minimum number of observed samples used for calculating periodical transfers. |
| Threshold for duration of a large burst (sec) | The duration threshold (in seconds) used for defining a large burst. |
| Threshold for size of a large burst (bytes) | The size threshold (in bytes) used for defining a large burst. |

The following device attributes are defined for the AT&T LTE profile.

| Device Attribute | Description |
|---|---|
| Device Name | The make and model of the device. |
| Average power during promotion (w) | The average power (in watts) used during promotion. |
| Average power of ping during short DRX (w) | The average power (in watts) used by a ping in the Short DRX state. |
| Average power of ping during long DRX (w) | The average power (in watts) used by a ping in the Long DRX state. |
| Average power during tail (baseline) (w) | The average power baseline (in watts) used in a tail state. |
| Average power of ping in idle (w) | The average power (in watts) used by a ping in the IDLE state. |

| | |
|---|---|
| Multiplier for throughput upload energy calc. (mW/Mbps) | The multiplier used for throughput energy calculations, expressed in mW/Mbps. |
| Multiplier for throughput download energy calc. (mW/Mbps) | The multiplier used for download energy calculations, expressed in mW/Mbps. |
| Baseline for CR energy (before throughput modifiers added) (w) | The baseline value (in watts) for energy used in the CR state, before throughput modifiers are added. |
| Average power for active GPS (w) | The average amount of power (in watts) for active GPS. |
| Average power for standby GPS (w) | The average amount of power (in watts) for standby GPS. |
| **Device Attribute** | **Description** |
| Average power when camera is on (w) | The average amount of power (in watts) when the camera is on. |
| Average power for active Bluetooth (w) | The average amount of power (in watts) for active Bluetooth. |
| Average power for standby Bluetooth (w) | The average amount of power (in watts) for standby Bluetooth. |
| Average power when screen is on (w) | The average amount of power (in watts) when the screen is on. |

v. AT&T WiFi Profile Network and Device Attributes

vi. The following network attributes are defined for the AT&T WiFi profile.

| Network Attribute | Description |
|---|---|
| Carrier | The network carrier for the device. |
| WiFi tail time (sec) | The amount of time (in seconds) spent in promotion from the IDLE state to the CR state. |
| Time delta for throughput calculations (sec) | The time delta (in seconds) used for calculating throughput. |
| Threshold for defining a burst (sec) | The time threshold (in seconds) used for defining a burst. |
| Threshold for defining a long burst (sec) | The time threshold (in seconds) used for defining a long burst. |
| Threshold for user input window (sec) | The time threshold (in seconds) used for calculating user input. |
| Min cycle for periodical transfers (sec) | The minimum cycle time (in seconds) used for calculating periodical transfers. |
| **Network Attribute** | **Description** |
| Max tolerable variation for periodical transfers (sec) | The maximum tolerable variation (in seconds) used for calculating periodical transfers. |
| Min observed samples for periodical transfers | The minimum number of observed samples used for calculating periodical transfers. |
| Threshold for duration of a large burst (sec) | The duration threshold (in seconds) used for defining a large burst. |
| Threshold for size of a large burst (bytes) | The size threshold (in bytes) used for defining a large burst. |

The following device attributes are defined for the AT&T WiFi profile.

| Device Attribute | Description |
|---|---|
| Device Name | The make and model of the device. |

| | |
|---|---|
| Average power WiFi connected (w) | The amount of inactive time (in seconds) spent in the CR state before changing to the DRX state. |
| Average power WiFi inactive (w) | The amount of time (in seconds) spent in the Short DRX state. |
| Average power for active GPS (w) | The average amount of power (in watts) for active GPS. |
| Average power for standby GPS (w) | The average amount of power (in watts) for standby GPS. |
| Average power when camera is on (w) | The average amount of power (in watts) when the camera is on. |
| Average power for active Bluetooth (w) | The average amount of power (in watts) for active Bluetooth. |
| **Device Attribute** | **Description** |
| Average power for standby Bluetooth (w) | The average amount of power (in watts) for standby Bluetooth. |
| Average power when screen is on (w) | The average amount of power (in watts) when the screen is on. |

### 5.1.4. Tools Menu

The Tools menu contains the following selections.

| Selection | Description |
|---|---|
| PCAP File Analysis | Opens Wireshark to access the Pcap file and displays the trace results in the Video Optimizer Data Analyzer. |
| Time Range Analysis | Displays the Time Range Analysis dialog box that allows you to set a time range, start or cancel the analysis, and display the analysis results. |
| Export html | Opens a Save As dialog box that allows you to export the currently loaded trace data to a html file. |

| | |
|---|---|
| Export Json | Opens a Save As dialog box that allows you to export the currently loaded trace data to a .json file. |
| Private Data Tracking | Opens a menu that allows you to set parameters for the Video Optimizer to search for in the trace. This allows you to find instances where private data is transmitted in an unsecure way. |
| Video Parser Wizard | Default settings for analysis of streaming video files downloaded during the trace. Settings can be overridden to customize the data to the specific application. |

#### *5.1.4.1. Time Range Analysis*

This feature allows you to set a time range, start or cancel the analysis, and display the analysis results. When the start and end time are entered and the Start button is clicked, the following results are displayed based on the network type of the Profile that has been selected.

For a 3G Profile, you will see the following results in the Time Range Analysis dialog box.

| Result | Description |
|---|---|
| Payload length | The length of the payload in bytes. |
| Total Bytes | The total number of bytes in the trace. |
| Energy | The amount of energy used in joules. |
| DCH Time | The amount of time, in seconds, that was spent in the DCH state. |
| Average Throughput | The average data throughput in kilobytes per second. |

For an LTE Profile, you will see the following results in the Time Range Analysis dialog box.

| Result | Description |
|---|---|
| Payload length | The length of the payload in bytes. |
| Total Bytes | The total number of bytes in the trace. |
| Energy | The amount of energy used in joules. |
| LTE Time | The amount of time, in seconds, that was spent in the Continuous Reception (CR) state. |
| Average Throughput | The average data throughput in kilobytes per second. |

For a WiFi Profile, you will see the following results in the Time Range Analysis dialog box.

| Result | Description |
|---|---|
| Payload length | The length of the payload in bytes. |
| Total Bytes | The total number of bytes in the trace. |
| Energy | The amount of energy used in joules. |
| WiFi Active Time | The amount of time, in seconds, that was spent in the WiFi Active state. |
| **Result** | **Description** |
| Average Throughput | The average data throughput in kilobytes per second. |

i. Export html:
The Export html feature creates an html document with the results from the Best Practices tab.

ii. Export Json:
The Export Json feature sends the statistical and analytical data from the current trace to a single. json file.

### 5.1.5. View Menu

The View menu contains the following selections.

| Selection | Description |
|---|---|
| Show Video Viewer | Opens a window that displays the video of the activities carried out on the device while the trace data was being collected. |
| Select Applications/IPs | Opens the Select Applications / IP Addresses dialog box that allows you to select the application and IP address that are included in the analysis. |
| Select Time Range | Opens a dialog to set a start and stop time. The trace will be re-analyzed between these two timestamps. |
| Options | Opens the View Options dialog box that allows you to select the events and states that will be plotted in the Diagnostics View chart |

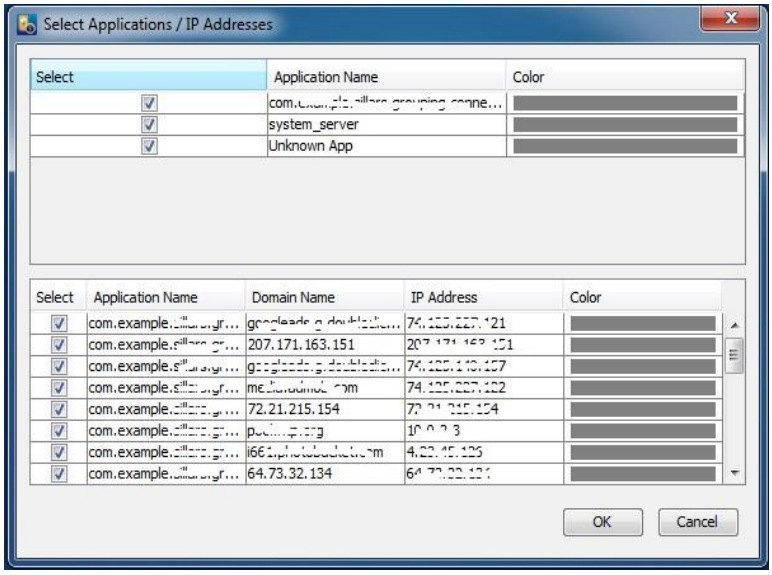
#### *5.1.5.1. Displaying a Video*

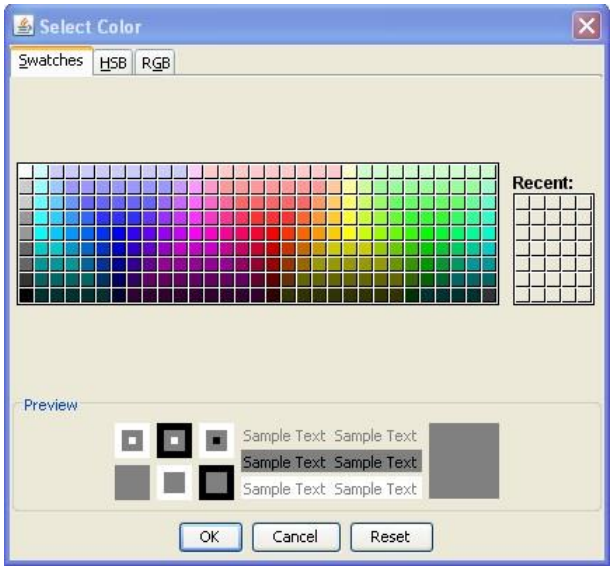
i. The Video Viewer in Video Optimizer displays the video of the activities carried out on the device while the trace data was being collected.

ii. Select Show Video Viewer from the View menu.
iii. Click any of the following controls at the bottom of the window.
iv. Play/Pause: Plays or pauses the video.
v. From 5 to 10 Step (Reverse): Steps one frame at a time in the reverse direction.
vi. Frame Step (Forward): Steps one frame at a time in the forward direction.
vii. Media Properties button: Displays the Media Properties dialog box with three tabs of information about the video: (General, Video, and
viii. Plug-in Settings) o General Tab: Displays the location of the media file, the content type, and the duration, the current position in the file, the bit rate, and the frame rate.
ix. Video Tab: Displays the encoding, size, and frame rate of the video.
x. Plug-in Settings Tab: Displays information about any additional plug-ins that are being used to display the video.

### *5.1.5.2. Selecting Applications and IP Addresses*

i. You can select individual applications from a table and assign colors to them. Another table lets you do the same thing to individual IP Addresses. The data from each selected application and IP Address is included in the analysis. The colors that you assign are used to mark the packet information for that specific application or IP Address on the Diagnostics Chart.
ii. Choose Select Applications/IPs from the View Menu.
iii. Click any of the rows in the Color column to select a color.



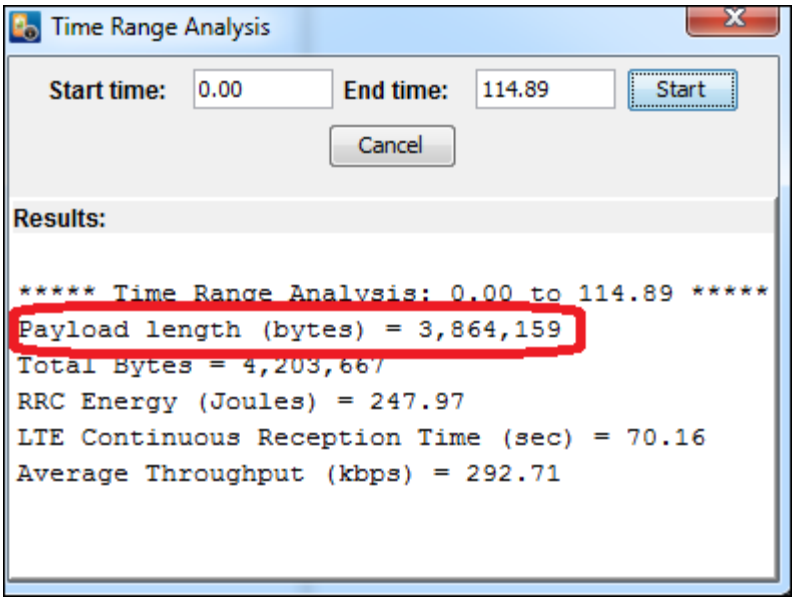



Once a color is selected, the trace will be recalculated, and all packets in the Diagnostics tab will be recolored based on your selections.

#### *5.1.5.3. Selecting a Time Range*

You can set a new time range for the trace analysis. This allows you to analyze a subset of the loaded trace. When you set the new start and end times, all of the analysis data in the Best Practices/Results, Overview, Diagnostics, and Statistics Tabs will display information for only your selected time range.

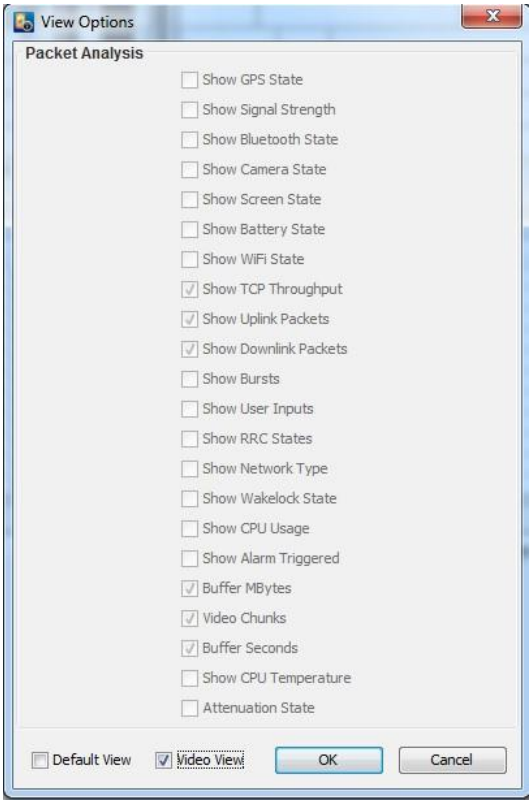
1. Choose Select Time Range from the View menu.
2. Enter a Start time and End time for the trace analysis.
3. Click OK.




#### *5.1.5.4. Options*

You can determine what will be plotted in the Diagnostics View chart. Default View and Video View are two checkboxes at the bottom with preset views for the diagnostic window. While the Video Optimizer will display all the options at once, we recommend keeping the list to under 11 views at a given time.

1. Select Options from the View menu.
2. Select the events and states that you want to be plotted in the Diagnostics View chart.
3. Click OK.




See the Diagnostics Tab section for more details on this menu.

### 5.1.6. Data Collector Menu

The Data Collector menu contains the following selections.

| Selection | Description |
|---|---|
| Start Collector | Starts the Video Optimizer Data Collector. |
| Stop Collector | Stops the Video Optimizer Data Collector. |

### 5.1.7. Help Menu

The Help menu contains the following selections.

| Selection | Description |
|---|---|
| FAQ | Opens the default web browser and displays the Video Optimizer FAQs web page. |
| User Guide | Opens the default web browser and displays the Video Optimizer User Guide |
| **Selection** | **Description** |

| | |
|---|---|
| Dependencies | Displays a dialog box containing the license information for the open source libraries and binaries that are distributed within the Video Optimizer package. |
| Forum | Opens the default web browser and displays the Video Optimizer User Forum web page. |
| Support | Provides a link to log in to the AT&T Developer Program and file a ticket with Video Optimizer Support. |
| Downloads | Opens the default web browser and displays the Video Optimizer Downloads web page that contains links for downloading and installing the different types of Video Optimizer. |
| Learn More | Opens the default web browser and displays the Learn More about Video Optimizer web page. |
| About | Displays a dialog box containing information about the Video Optimizer application including its version. |

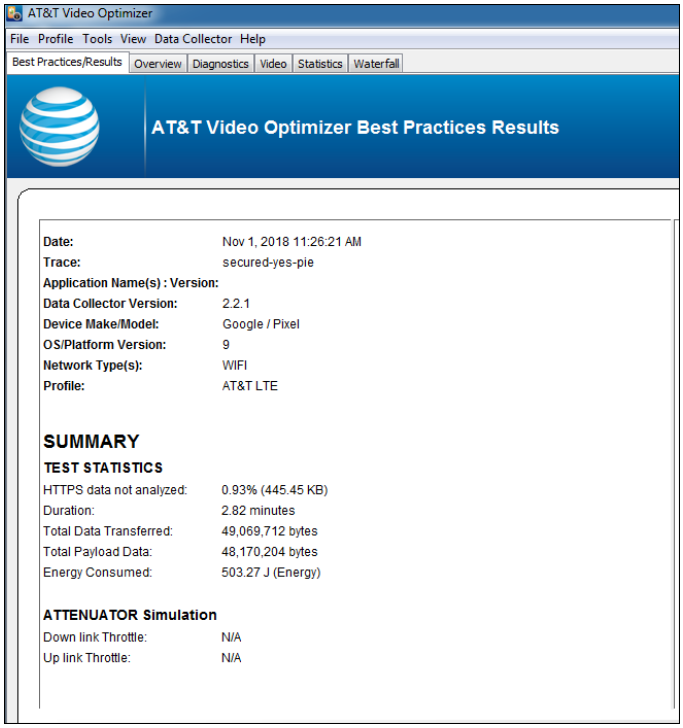
## 5.2. Content Tabs

The Video Optimizer user interface is divided into the following tabbed sections.

| Tab | Description |
|---|---|
| Best Practices / Results | Displays the results of the Best Practices tests that are conducted on the trace data. |
| Overview | Displays charts and tables that present an overview of key statistical data from the trace. |
| Diagnostics | Displays charts and tables that present key diagnostic data from the trace. |
| Video | Displays Video Results Summary, Chart and Movie Manifests, Video Requests. |
| Statistics | Displays charts and tables that present key statistical data from the trace. |
| Waterfall | Displays a waterfall view chart of the TCP connections from the trace spread over time. The details, requests, and responses of each connection can be viewed when the chart is clicked. |

### 5.2.1. Best Practices/Results Tab

The Best Practices/Results tab displays the results for all the Best Practices tests that are conducted on the data captured in the trace files.




When trace files are loaded into Video Optimizer, The Best Practices/Results tab displays the following test results.

- A header section with basic information about the trace.
- A SUMMARY section with high-level test statistics and attenuator information.
- A TESTS CONDUCTED section that lists all the tests and has a basic pass/fail/warning result indicated by a specific icon.
- High-level results pages for the all the tests grouped into the test categories: File Download, Connections, HTML, Security, Video, and Others.

Each of these pages has a common header panel that shows the following information about the loaded trace files:

| Label | Description |
|---|---|
| Date | The date when the trace files were generated. |
| Trace | The name of the folder containing the trace files. |
| Application Name (s): Version | The names and versions of the applications that were running when the trace data was collected. |
| Data Collector Version | The version of the Video Optimizer Data Collector that was used to collect the trace data. |

| | |
|---|---|
| Device Make/Model | The make and model of the device from which the data was collected. |
| OS/Platform Version | The operating system version or platform version of the device that the trace was captured on. |
| Network Type(s) | The types of any networks (such as 3G or LTE) that were in use when the data was collected. |
| **Label** | **Description** |
| Profile | The device profile that was used for the trace analysis. |

#### *5.2.1.1. iPhone Make/Model*

The iPhone make & model displayed by Video Optimizer is confusing to many users. However, it is based on the manufacturer settings. Please refer to the chart below to clarify iOS make & model.

| VO displays | iOS device | VO displays | iOS device |
|---|---|---|---|
| iPhone1,1 | iPhone | iPhone8,3 | iPhone SE (GSM+CDMA) |
| iPhone1,2 | iPhone 3G | iPhone8,4 | iPhone SE (GSM) |
| iPhone2,1 | iPhone 3GS | iPhone9,1 | iPhone 7 |
| iPhone3,1 | iPhone 4 | iPhone9,2 | iPhone 7 Plus |
| iPhone3,2 | iPhone 4 GSM Rev A | iPhone9,3 | iPhone 7 |
| iPhone3,3 | iPhone 4 CDMA | iPhone9,4 | iPhone 7 Plus |
| iPhone4,1 | iPhone 4S | iPhone10,1 | iPhone 8 |
| iPhone5,1 | iPhone 5 (GSM) | iPhone10,2 | iPhone 8 Plus |
| iPhone5,2 | iPhone 5 (GSM+CDMA) | iPhone10,3 | iPhone X Global |
| iPhone5,3 | iPhone 5C (GSM) | iPhone10,4 | iPhone 8 |
| iPhone5,4 | iPhone 5C (Global) | iPhone10,5 | iPhone 8 Plus |
| iPhone6,1 | iPhone 5S (GSM) | iPhone10,3 | iPhone X |
| iPhone6,2 | iPhone 5S (Global) | iPhone10,6 | iPhone X GSM |

| | | | |
|---|---|---|---|
| iPhone7,1 | iPhone 6 Plus | iPhone11,2 | iPhone XS |
| iPhone7,2 | iPhone 6 | iPhone11,4 | iPhone XS Max |
| iPhone8,1 | iPhone 6s | iPhone11,6 | iPhone XS Max China |
| iPhone8,2 | iPhone 6s Plus | iPhone11,8 | iPhone XR |

#### *5.2.1.2. Summary*

The Summary section shows the summary of results for all the best practices tests conducted on the loaded trace files. It contains the following sections.

TEST STATISTICS shows the following information about the loaded trace files.

| Label | Description |
|---|---|
| HTTPS data not analyzed | The percentage of total content and the size of content, in KB, that was downloaded over HTTPS and was not analyzed. |
| Duration | The total time, in minutes, for which the trace data was collected. |
| Total Data Transferred | The total size, in bytes, of all data packets that are transferred for the entire duration of the trace data collection. This total includes the size of the packet and the packet header. |
| Total Payload Data | This gives the total payload of the entire traffic. |
| Energy Consumed | The total energy, in Joules, that is consumed during the entire duration of the trace data collection. This total includes the energy of RRC, GPS, WiFi, Bluetooth, Camera and Screen. |

ATTENUATOR shows the parameters that were used for network attenuation during collection. For example, Downlink throttle provides the ms delay in the downlink throttling, and uplink is the delay placed on uplink connections. See Network Attenuation.

TESTS CONDUCTED displays a list of all the best practices tests that were conducted on the loaded trace files with an icon to the left of each test name that indicates the test

result status (Pass, Fail, Warning, Configuration required and no data(N/A) icon.

The tests are grouped into the following categories.

**Note**: When any of the best practice tests in a category fails, the header for that category turns red. The category header remains green if all the tests in that category have passed.

| Test | Category | Description |
|---|---|---|
| Text File Compression | File Download | Tests if any text files sent by the app that were larger than 850 bytes are uncompressed. |

| Test | Category | Description |
|---|---|---|
| Duplicate Content | File Download | Tests if more than three files are downloaded in a duplicate manner in the loaded trace files. |
| Cache Control | File Download | Tests if the amount of “not expired duplicate data” is greater than the amount of “not changed data” in the loaded trace files. |
| Content Expiration | File Download | Tests if there is more than 10% of non-cacheable data available in the loaded trace files. |
| Combine Java Script and CSS Requests | File Download | Tests if there are multiple requests for CSS or JS files occurring within 2 seconds of one another. |
| Resize Images for Mobile | File Download | Tests if there are any images that are 150% larger than the area specified for them. |
| Image Metadata | File Download | Tests for EXIF text metadata in your images. An image fails if the file is more than 1% metadata. |

| | | |
|---|---|---|
| Image Compression | File Download | Tests JPEG images for compression. If a file is saved at 85% quality and is >15% smaller (in KB), the best practice fails. |
| Image Format | File Download | Alternative image formats can help increase the speed of your mobile application, as newer compression algorithms compress the files to a much smaller size, with virtually no loss in quality. |
| Image Comparison | File Download | Images downloaded from the network should be approximately the same size as the images that appear on the screen. Video Optimizer measures each image on the screen and compares the dimensions to the images downloaded over the network. |
| Minify CSS, JS, JSON and HTML | File Download | Tests if there are any files that could be minified (shrunk through the removal of whitespace). |
| CSS Sprites for Images | File Download | Tests for any groups of small images that are downloaded at once, which could be combined into one image using sprites. |
| Unnecessary Connections – Multiple Simultaneous Connections | Connections | Tests if there are several bursts in a row that are not user initiated in the loaded trace files. |

| Test | Category | Description |
|---|---|---|

| | | |
|---|---|---|
| Multiple Simultaneous Connections to One Endpoint | Connections | With HTTP2, you can utilize multiplexing – to download multiple files in one connection. Using fewer connections allows content to be downloaded faster and uses fewer resources on the device. |
| Multiple Simultaneous Connections to Many Endpoints | Connections | Opening many connections all at once can cause bottlenecks in slower network conditions. Android limits the number of TCP connections to 15. |
| Inefficient Connections – Periodic Transfers | Connections | Tests if a periodic connection is detected in the loaded trace files. |
| Inefficient Connections – Screen Rotation | Connections | Tests if the application triggers network activity when the screen orientation changes. |
| Inefficient Connections – Connection Closing Problems | Connections | Tests if 5% of the energy is used for TCP control in the loaded trace files. |
| 400,500 HTTP Status Response Codes | Connections | Tests if there are any HTTP response codes in the 400 range (indicating a client request error) or in the 500 range (indicating a server request error) in the loaded trace files. |

| | | |
|---|---|---|
| 301,302 HTTP Status Response Codes | Connections | Tests if there are any occurrences of the HTTP status response code 301 (indicating that the URI of a requested resource has been changed permanently), and any occurrences of the HTTP status response code 302 (indicating that the URI of a requested resource has been changed temporarily) in the loaded trace files... |
| 3rd Party Scripts | Connections | Tests for files where at least 2 external scripts are being called. |
| Asynchronous Load of JavaScript in HTML | HTML | Tests for any HTML files with a synchronous load of JavaScript in the HEAD. |

| Test | Category | Description |
|---|---|---|
| Http 1.0 Usage | HTML | Tests if HTTP 1.0 is seen in the header of the loaded trace files. |
| File Order | HTML | Tests for any HTML files where JS is loaded immediately before CSS. |
| Empty Source and Link Attributes | HTML | Tests for the empty attributes: iframe src, href src, img src, script src, and link href, in the trace files. |
| FLASH | HTML | Tests for any references to the Flash player in the loaded trace files. |
| “display: none” in CSS | HTML | Tests for any instances of the CSS command “display:none” in the trace files. |
| Security: HTTPS Usage | Security | Looks for connections that do not feature HTTPS. |

| | | |
|---|---|---|
| Transmission of Private Data | Security | Scans all transmitted data for potential leaks of private data. |
| Unsecure SSL Version | Security | There are several versions of SSL that are not considered secure. This test identifies connections that use those versions. |
| Weak Cipher | Security | Scans for security with weak ciphers that can be easily compromised. |
| Forward Secrecy | Security | Encryption that allows key capture, allowing attackers to gain access to all security keys. |
| Stalls | Video | Identifies when a video stalls (stops playing due to an empty buffer). |
| Start Up Delay | Video | Quantifies the time it takes for video to start from the first download of video data. |
| Buffer Occupancy | Video | Quantifies the amount of video stored in the buffer over time. |

| Test | Category | Description |
|---|---|---|
| Network Comparison | Video | Compares the video bandwidth download with the available network bandwidth, to determine if the optimal bitrate is being displayed. |
| TCP Connection | Video | Count of connections used to deliver video during the trace. |
| Segment size | Video | Average size of the video chunks downloaded during streaming. |
| Segment Pacing | Video | Pacing of the video chunks downloaded during streaming. |
| Redundancy | Video | Looks to see if the same chunk is downloaded multiple times. |

| | | |
|---|---|---|
| Concurrent Session | Video | Using too many concurrent sessions may cause problems with smooth delivery of content |
| Variable Bitrate | Video | Analysis for Non- DRM stream. |
| Resolution and Perception | Video | Video Optimizer conducts a test that flags video files with resolutions greater than 720p that are being sent to a smartphone. |
| Accessing Peripheral Applications | Others | Tests if any peripheral applications are seen to be ON for more than 5% of the total duration recorded in the loaded trace files. The peripheral applications checked during this test are: GPS, WiFi, Bluetooth, and camera. |

For more detail on each of these best practices, visit https://developer.att.com/video-optimizer/docs/best-practices.

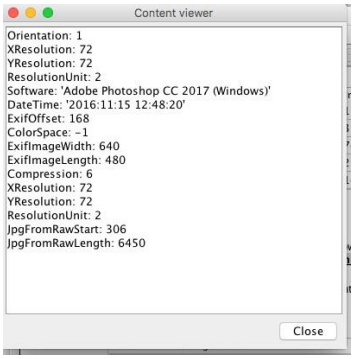
#### *5.2.1.3. Tables*

The tables in the Video Optimizer best practices sections help you pinpoint the issues flagged by the best practice and they let you further investigate the issues.

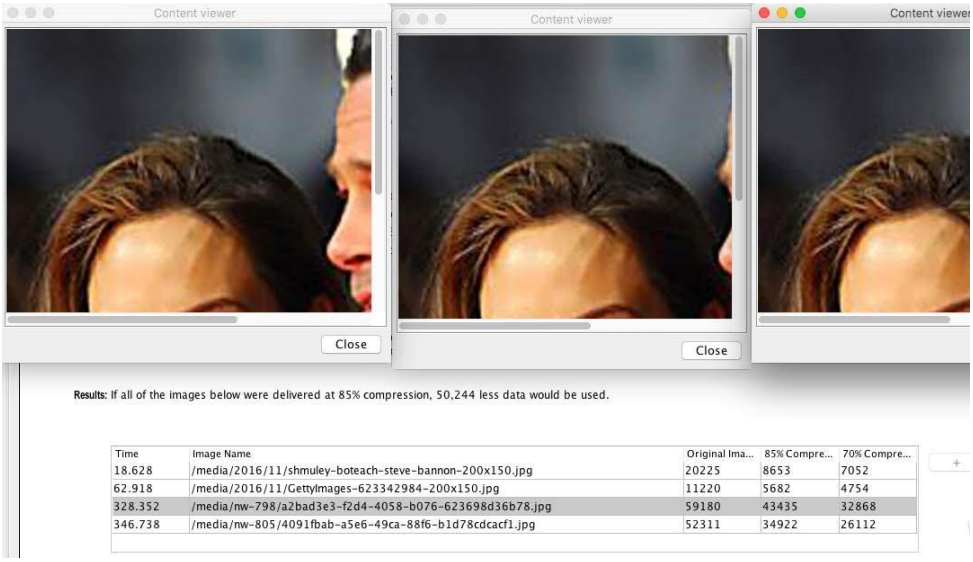
1. Double-click a line in the Best Practices table to view the Diagnostics tab, which highlights the connection or file that is failing the best practice.
2. Right click a line in the Best Practices table to export the table as csv. (Alternatively, command-a (select all) command-c (copy) will allow you to paste (command v) the table into your favorite spreadsheet software.

Images tables have additional features such as metadata views.

1. Click an image table to open a content view with the metadata:




2. Click the file name or original file size columns in the Image Compression table to open a view of the original image.
3. Click the 85% column to open the 85% compressed image and click the 70% column to open the 70% compressed image. In **Figure**, all three images are open in the Content Viewer (the original image, the 85% compressed image, and the 70% compressed image).




### 5.2.2. Overview Tab

The Overview Tab displays charts and tables that summarize the data in the loaded trace files. The top part of the Overview tab contains the following information:

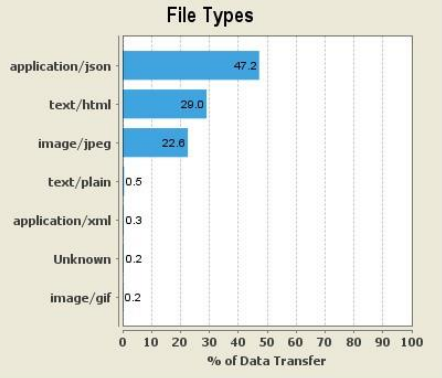
- Date: The date when the trace files were generated.
- Trace: The name of the folder containing the trace files.
- Downlink Throttle: If the Network attenuator downlink function was used during this trace, it will register the speed in mbps/kbps.

- Uplink Throttle: If Network Attenuator uplink function was used during this trace, this will record the speed in mbps/kbps.
- Network Type: The type of network, like 3G or LTE, which was in use when the data was collected.
- Profile: The profile that was used for the trace analysis.
- Total Bytes: The total number of bytes in the trace

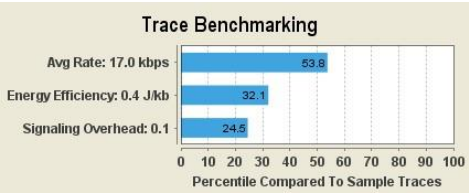
#### *5.2.2.1. Charts*

The top of the Overview tab has three charts with high level information about the type of content, the energy usage compared with benchmark traces, and the number and type of connections used.

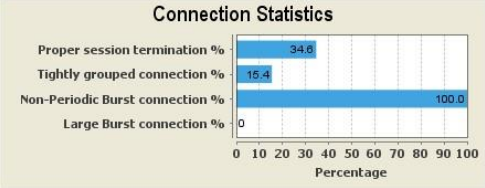
The File Types chart plots the percentage of the various file types found in the trace data.




The Trace Benchmarking chart plots the average data rate, energy efficiency, and signaling overhead of the loaded trace, as a percentage, compared with a set of sample benchmark traces.




The Connection Statistics chart plots the percentage of the various types of session terminations based on the data captured in the trace. Four types of session terminations are plotted: Proper session termination, tightly grouped connection, periodic bursts connection, and large burst connection.




The Connection Statistics chart contains the following fields:

| Field | Description |
|---|---|
| Proper session termination | If the amount of time between the last data packet and the data packet that signaled the TCP session termination is less than or equal to 1 sec, then the session termination is represented as proper session termination. In the session termination plot, the sessions are displayed as a percentage of the total number of TCP sessions. |
| Tightly grouped connection | If 3 bursts occur in less than 15 seconds or 4 bursts occur in less than 60 seconds, then those sets of bursts are referred to a tightly grouped connection. In the session termination plot, these bursts are displayed as a percentage of the total number of bursts. |
| Non-Periodic Burst connection | If the Internet Addresses, host names, or object names are not the same for the packets in a set burst over a period, then those bursts are considered nonperiodic bursts. In the session termination plot, these bursts are displayed as a percentage of the total number of bursts. |
| Large Burst connection | If the burst duration is more than 5 seconds, then that burst is considered a large burst. In the session termination plot, these bursts are displayed as a percentage of the total number of bursts. |

### *5.2.2.2. Tables*

The bottom section of the Overview tab has tables for duplicate content, accessed domains, and the TCP sessions in the accessed domains. You can get more information about these items by clicking on table rows.

The Duplicate Content table lists files that have been identified as duplicate content. By default, the table is sorted by content type, and it includes the following columns.

| Column | Description |
|---|---|

| | |
|---|---|
| Duplicate Content Type | One of the following types of duplicate content:<br>ORIGINAL_FILE<br>OBJDUP_NOT_EXPIRED<br>OBJDUP_NOT_CHANGED_SERVER<br>OBJDUP_NOT_CHANGED_CLIENT<br>OBJDUP_PARTIAL_NOT_CHANGED_SERVER<br>OBJDUP_PARTIAL_NOT_CHANGED_CLIENT<br>OBJDUP_PARTIAL_NOT_EXPIRED |
| Time | The timestamp for this occurrence of the duplicate content. |
| File Name | The name of the duplicate file. |
| File Size (bytes) | The size of the duplicate file in bytes. |

1. Click the title of any column to sort the table.
2. Select a file and click View or Save As to view or save the content of the file.
3. Right-click the table to export the data in CSV format.

The Accessed Domains table contains details about each domain that was accessed during the trace. The table includes the following columns.

| Column | Description |
|---|---|
| Domain Name | The list of domain names that are captured in the loaded trace files. These domain names are application independent and may have occurred in the browser app or any other application. |
| TCP Sessions | The count of TCP sessions for the corresponding domain name. |
| Average Session Length (sec) | The average session length in seconds. This average is calculated by dividing the total TCP session time (the difference between the session end time and the session start time) by the size of the session for this domain name. |
| Files Downloaded | The number of files downloaded for the domain name session. |

1. Click the title of any column to sort the table.
2. Left-click a domain in this table to populate the adjoining table (Domain TCP Sessions) with information about the TCP sessions that were used when accessing that domain.
3. Right-click the table to export the data in CSV format.

The Domain TCP Sessions table contains the collection of TCP Session information for the currently selected domain name in the Accessed Domains table and refreshes each time a new domain name is selected. The table includes the following columns.

| Column | Description |
|---|---|
| Time | The time stamp of the Domain TCP Session. |
| Remote IP Address | The Remote IP Address of the Domain TCP Session. |
| Local Port | The Local port value of the Domain TCP Session. |
| Session Length (sec) | The session length, in seconds, of the Domain TCP Session. The session length is the difference between the starting time stamp and the ending time stamp for the session. |
| Bytes Transmitted | The number of bytes transmitted during the Domain TCP session. |
| Session Close Delay (sec) | The session termination delay, in seconds, of the Domain TCP Session. |
| Closed By | Indicates whether the Client or the Server closed the Domain TCP Session. The Closed By value can be Client, Server, or Status Unknown, and is determined by the session packet direction. |

1. Click the title of any column to sort the table.
2. Double-click a domain in this table to navigate to the TCP/UDP Flows Table in the Diagnostics tab where you can view the TCP flow information for the selected Domain TCP session. The selected TCP information will be indicated by highlighted type.
3. Right-click the table to export the data in CSV format.

### 5.2.3. Diagnostic Tab

The Diagnostics tab plots data from the loaded trace files and displays it in a Diagnostics chart. Detailed information is displayed in the TCP/UDP Flows Table, which has additional tabs for three different views into the TCP or UDP data.


The Diagnostics tab displays the following information at the top before the chart:

- Date: The date when the trace files were generated.
- Trace: The name of the folder containing the trace files.
- Downlink Throttle: If the Network attenuator downlink function was used during this trace, it will register the ms delay.
- Uplink Throttle: If Network Attenuator uplink function was used during this trace, this will record the ms delay uplink.
- Total Bytes: The total number of bytes in the trace.
- Network type: The type of network, like 3G or LTE, which was in use when the data was collected.
- Profile: The profile that was used for the trace analysis.

#### *5.2.3.1. Diagnostics Tab Chart*

The Diagnostics tab chart displays the graph data with an X-axis for the trace timeline and a Y-axis for labels of the information that is being plotted.

Click  or  for a zoom view.

Click  to save a snapshot of the chart.

Use the following procedure to configure the items that are plotted on the Diagnostics tab chart.

1. Select Options from the View menu.
2. Select the items to be plotted in the View Options dialog
3. Click OK.
4. Move the pointer over the graphed items in the Diagnostics tab chart for more information on each one.


**Note**: In addition to creating your own custom view, there is a Default View and a Video View – especially created to help analyze video traces.

### *5.2.3.2. Diagnostics Tab Chart Items*

The following sections describe in detail the different types of information that can be plotted on the Diagnostics tab chart.

- GPS State— shows the variation in GPS states over the duration of the trace.

| GPS State | Description |
|---|---|
| GPS Active | The GPS receiver is turned on and is fixing the location. Energy consumption during this state is high. This state is colored green in the plot. |
| | The GPS receiver is turned on but is in standby mode. Energy consumption during this state is low. This state is colored yellow in the plot. The GPS receiver is turned off. |

| | |
|---|---|
| | |

- Signal Strength—shows the variation in radio signal strength (expressed in Dbm) over the duration of the trace.
- Bluetooth State—shows the variation in Bluetooth states over the duration of the trace.

| Bluetooth State | Description |
|---|---|
| Bluetooth Connected | The Bluetooth is turned on and the device is paired with another device for data transfer. Energy consumption during this state is high. This state is colored green in the plot. |
| Bluetooth Standby | The Bluetooth is turned on, but the device is not paired with another device. Energy consumption during this state is low. This state is colored yellow in the plot. |
| Bluetooth Off | The Bluetooth is turned off. |

- Camera State—shows the variation in camera states over the duration of the trace

| Camera State | Description |
|---|---|
| Camera On | The Camera is turned on. Energy consumption during this state is high. This state is colored green in the plot. |
| Camera Off | The Camera is turned off. |

- Screen State—shows the variation in screen states over the duration of the trace.

| Screen State | Description |
|---|---|
| Screen On | The Screen is active. Energy consumption during this state is high. This state is colored green in the plot. When you place the tooltip over this state in the plot, the screen time out value (in seconds) and the brightness (in %) are displayed. |
| Screen Off | The device is in sleep mode. |

- Battery State—shows the variation in battery level over the duration of the trace. Includes battery level, battery temperature, and battery connection status.
- WiFi State—shows the variation in WiFi states over the duration of the trace.

| WiFi State | Description |
|---|---|
| WiFi Connecting | The device is trying to connect to a WiFi network. Energy consumption during this state is high. This state is colored green in the plot. |
| WiFi State | Description |
| WiFi Connected | The device is connected to a WiFi network. Energy consumption during this state is high. This state is colored green in the plot. In addition to state detail, also shows the Mac Address, Radio Received Signal Strength Indication (RSSI) and Service set identifier (SSID). |
| WiFi Disconnecting | The device is disconnecting from a WiFi network. Energy consumption during this state is high. This state is colored green in the plot. |
| WiFi Standby | The device is disconnected from WiFi network. Energy consumption during this state is low. This state is colored yellow in the plot. |
| WiFi Suspended | A WiFi network was disconnected unexpectedly. Energy consumption during this state is |

| | |
|---|---|
| | low. This state is colored yellow in the plot. |
| WiFi Off | WiFi is disabled in the device. |
| WiFi Unknown State | The WiFi is in an unknown state. |

- TCP Throughput (Throughput)—shows the variation in network traffic (expressed in kbps) over the duration of the trace. The higher the throughput, the higher the energy consumption.
- Uplink Packets (Packets UL)—shows the packets that were uploaded (uplinked) in the TCP sessions over the duration of the trace. Includes Packet Info, TCP Session Info, and HTTP Info.
- Downlink Packets (Packets DL)—shows the packets that were downloaded (down linked) in the TCP sessions over the duration of the trace. Includes Packet Info, TCP Session Info, and HTTP Info.
- Bursts—shows the various types of bursts that occurred over the duration of the trace. A burst consists of consecutive packets transferred in a batch. Includes the burst type, a message, the packet count, the total bytes, and the throughput (in kbps).

| Burst Type | **Description** |
|---|---|
| TcpControl | This category of burst is colored blue in the chart plot, and displays the tooltip message: ”TcpControl: Traffic that is delayed from a previous burst.” |
| TcpLossRecoverO rDup | This category of burst is colored black in the chart plot, and displays the tooltip message: “TcpLossRecover: Traffic that has been resent due to long delay.” |
| UserInput | This category of burst is colored green in the chart plot and displays the tooltip message: “User Input: Traffic initiated after a User Input event.“ |
| **Burst Type Description** | |
| Screen Rotation | This category denotes a burst caused by the rotation of the device. It displays the following tooltip message: “Screen Rotation: This traffic was initiated by a rotation of the device”. |

| | |
|---|---|
| App | This category of burst is colored red in the chart plot and displays the tooltip message: "App: Traffic initiated by the client." |
| SvrNetDelay | This category of burst is colored yellow in the plot, and displays the tooltip message: "SvrNetDelay: Traffic initiated by the server." |
| Large Burst | If a burst duration is more than 5 seconds, then that burst is considered to be a long (or large) burst. This category of burst is colored gray in the chart plot, and displays the tooltip message: "LargeBurst: Traffic in a large burst (configurable in settings)." Note: The tooltip refers to the fact that the length and size thresholds for what is considered to be a long/large burst can be configured using the Customize dialog in the Profile menu. |
| Periodical | If the Internet Addresses, or the host names, or object names are the same for the packets in a set burst over a period of time, then those bursts are considered periodic bursts. This category of burst is colored purple/pink in the chart plot, and displays the tooltip message: "Periodical: Traffic that has a distinct periodic pattern." |
| User Defined | These are user defined bursts. This category of burst is colored magenta in the chart plot. |
| Unknown | This category denotes an unknown type of burst. |

- User Input—shows the various user input events that have occurred over the duration of the trace including the following:

  ○ Screen Touch ○ Power Button ○ Volume Up ○ Volume Down

  ○ Ball Key ○ Home Key ○ Menu Key ○ Back Key ○ Search Key Green Key ○ Red Key ○ Key Press/Screen Touch ○

Screen Orientation Changed to Landscape Screen

Orientation Changed to Portrait ○ Unknown event

- RRC States—shows the variation in the Radio Resource Control (RRC) states over the duration of the trace. The states are determined by calculating the battery usage when network packets are received.

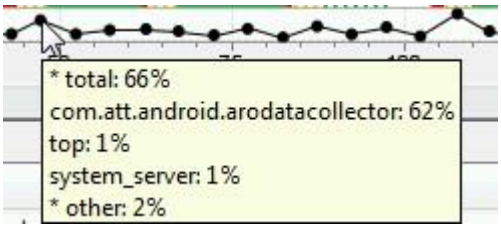
  **Note:** DCH stands for dedicated channel, FACH stands for forward access channel, and CR stands for Continuous Reception.

| RRC State | Description |
|---|---|
| IDLE | Indicates the radio is off. |
| DCH (Active) | This state is colored yellow in the chart plot. It indicates that the radio is in a high data, high radio energy, and high bandwidth mode which allows maximum throughput. |
| DCH TAIL | This state is colored with a yellow cross hatch pattern in the chart plot. It indicates that the radio is in a high throughput, high bandwidth state, but no packets are being sent. |
| FACH (Standby) | This state is colored green in the chart plot. It Indicates that the radio is in low power state. Signaling packets may be sent, but content requires transition to DCH. |
| FACH TAIL | This state is colored with a green cross hatch pattern in the chart plot. It indicates that the radio is in a low power state with no traffic. |
| PROMOTION IDLE->DCH (Active) | Transition from IDLE to DCH (Active) state. This state is represented by a red triangle in the chart plot. It indicates the radio switching from off to a high-power state. |
| PROMOTION FACH (Standby)->DCH (Active) | Transition from FACH (Standby) to DCH (Active) state. This state is represented by a red polygon in the chart plot. It indicates switching from low power state to the high-power state. |

| | |
|---|---|
| LTE IDLE | Indicates that the radio is in an idle state, with occasional pings to the network for data. |
| LTE PROMOTION | Transition from IDLE to the Continuous Reception state. This indicates the time and power associated with the radio switching from idle to active. |
| RRC State Description | |
| LTE CONTINUOUS | Continuous Reception is the time of active packet transfer. High throughput high energy data transfer. Energy here is indicated as constant, but it does vary based on throughput. |
| LTE CR TAIL | Continuous Reception Tail is the Inactivity timer after packets are sent, prior to DRX. |
| LTE DRX SHORT | The Short DRX state indicates that the radio is in a high bandwidth, high energy state, looking for packets. |
| LTE DRX LONG | The Long DRX state is the LTE Tail. It indicates that the radio is in a high bandwidth, high energy state, looking for packets. |
| WIFI ACTIVE | The WiFi transmitter is at full power - sending and receiving information. |
| WIFI TAIL | The WiFi Tail state is an inactivity timer after packets are sent. |
| WIFI IDLE | The Radio is in an idle state, with a small trickle of power usage |

- Network Type—indicates how long the device has been connected to a particular network type, or if the network type has changed during the trace. The network types that are identified include GPRS, UMTS, HSDPA, HSPA, HSPAP, HSUPA, and LTE.
- Wakelock State—indicates whether the device is in the wake state (active for user input) or the lock state during the trace. Includes the number of times the wakelock state has been changed from the plot point to the beginning of the trace.
- CPU Usage—shows the total percentage of CPU usage for each process followed by the name of each selected process

and the percentage of CPU that it is using (**Figure**). Although the CPU Usage percentages for all the processes can be viewed at once, this chart is most useful when only one process is selected or when a small number of processes are selected, because the changes between the data points is more clearly visible. See 5.1.4.4 Selecting Processes.




**Note:** It is possible that the percentages for individual processes may not always add up to the total percentage. This is because the process called top is a command that pulls the information from different locations that may not be completely in sync.

- Alarm Triggered—indicates when an application has triggered an alarm. Includes the type and timestamp of the alarm, the name of the application that triggered it, and the number of times it repeated.
- Buffer Occupancy—provides an estimate on how much video (in KB) has been downloaded into the buffer. This line is populated after the video chunk startup is calculated (see Video analysis section for details).
- Video Chunks—displays the first frame of each chunk (for videos without DRM) to indicate when the chunk download started. If the chunk has DRM, or Video Optimizer cannot parse the first frame, a placeholder image will be displayed.

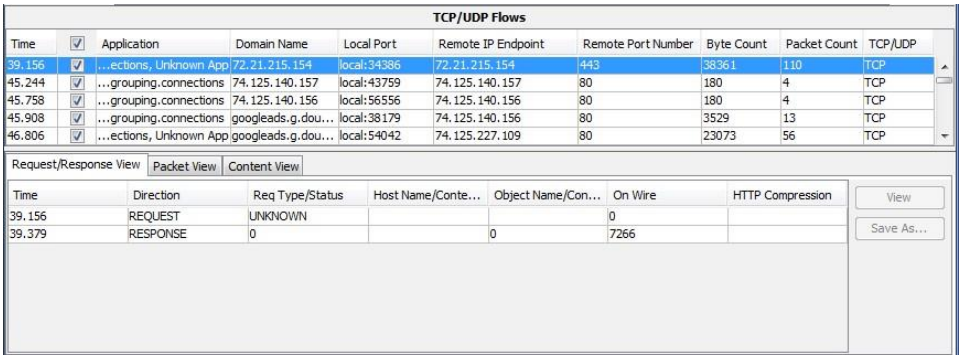
1. Click a chunk thumbnail to open a dialog to “match” the screenshot with the time it appears in the video (see video analysis section for details).
2. Once the video has been matched with the screen video, point to a video frame to display information about the video playback.

- Playback Buffer Occupancy—provides an estimate on how much video (in seconds) is stored in the buffer. This line is populated after the video chunk startup is calculated (see Video analysis section for details).

### *5.2.3.3. Diagnostics Tab Tables*

The TCP/UDP Flows table in the Diagnostics tab has rows of TCP or UDP session data that correspond to trace points in the Diagnostics chart.

1. Select or deselect the rows of TCP or UDP data to narrow the trace data that appears in the Diagnostics chart.
2. Click the check box in the title row to select or deselect *all* of the TCP and UDP data.

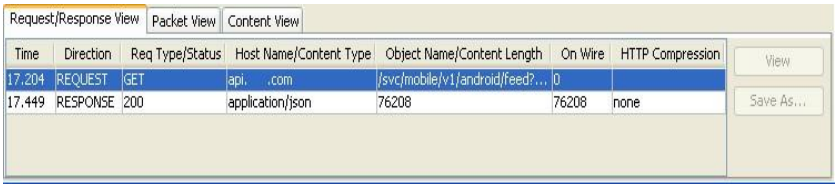



The TCP/UDP Flows table contains the following columns of information:

| Column | Description |
|---|---|
| Time | The amount of time (in seconds) from the beginning of the trace, when this request was made. |
| Application | The name of the client application that is making the request. |
| Domain Name | The name of the domain to which the request is being made. |
| Local port | The local port through which the request is being made |
| Remote IP Endpoint | The IP address of the domain on which these requests are being listened for. |
| Remote Port Number | The number of the remote port (on the domain) through which the request is being received. |
| Byte Count | The number of bytes in the TCP/UDP flow. |
| Packet Count | The number of packets in the request. |
| TCP/UDP | Indicates whether the TCP or UDP or quick or DNS protocol was used. |

A second table below the TCP/UDP Flows table has tabs that let you change the data view in three different ways: Request/Response (default view), Packet, or Content View.

The Request/Response View tab displays the request/responses associated with the selected row highlighted in the TCP/UDP Flows Table.

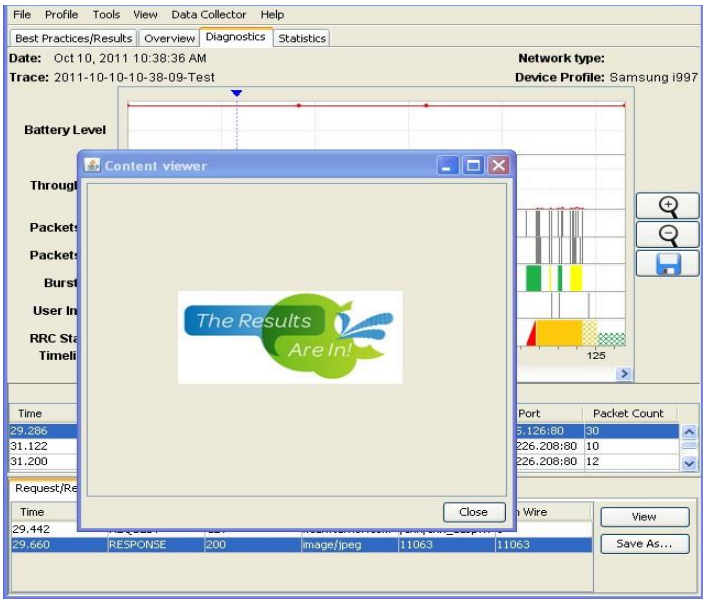


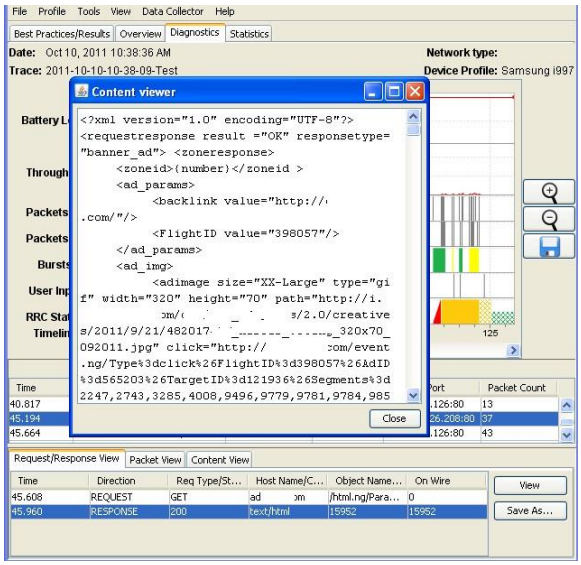

Column headings in the Request/Response View have different meanings (separated by a "/" in the column name) depending on whether the row contains a request or a response.

| Column | Description |
|---|---|
| Time | The time of the request or response, in seconds, from the beginning of the trace. |
| Direction | The direction of the TCP flow (REQUEST or RESPONSE) |
| Req Type/Status | HTTP request type (GET, PUT, POST, or DELETE) or the HTTP status of the response (such as 200 for OK or 404 for resource not found). |
| **Column** | **Description** |
| Host Name/Content Type | The host name for the HTTP request or the content type of the response. Content type consists of a pair of values (type/subtype) representing the Internet media type, for example, text/plain (simple text messages), text/html (HTML document), text/CSS (cascading style sheet), image/gif (GIF Image), image/jpeg (JPEG Image), application/JSON (JSON data object). |
| Object Name/Content Length | The name of the object requested from the host or the length, in bytes, of the response. |
| On Wire | The number of bytes on the wire during this request or response. |
| HTTP Compression | This column is only used for responses with a text MIME type. If the response contains a text file, this column indicates if HTTP compression was used or if there was none. |

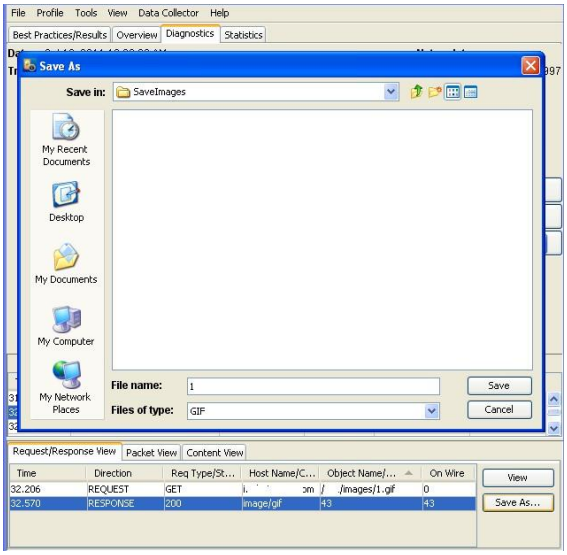
1. Select one of the rows from the Request/Response View tab that has a response.

2. Click View to display the data object in the response. For an image, the image will be displayed in the Content viewer (**Figure**). If the data object is text, HTML, or JSON it will be displayed in the Content viewer (**Figure**).



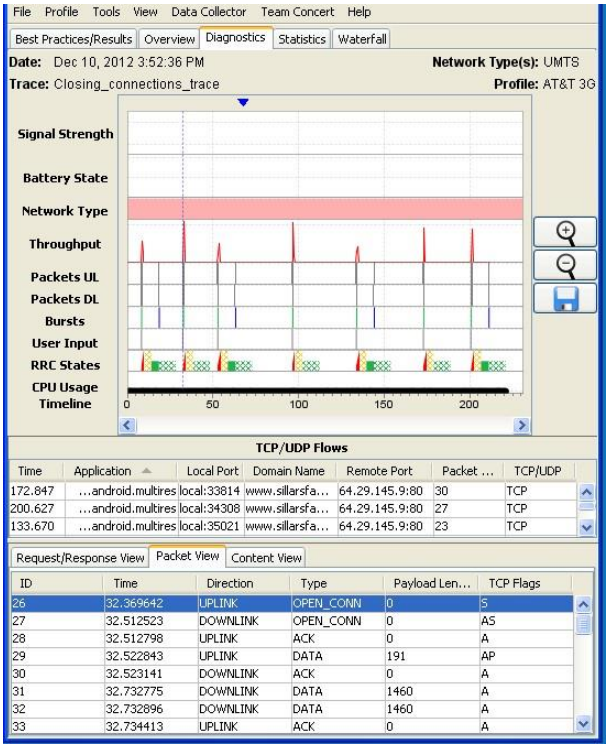




3. Click Save As to save the object as a file in the specified directory.
(Note that the View and Save As buttons are not available for Requests).




If the object cannot be displayed, an error message will indicate that the content was unable to be viewed because it may be corrupted.

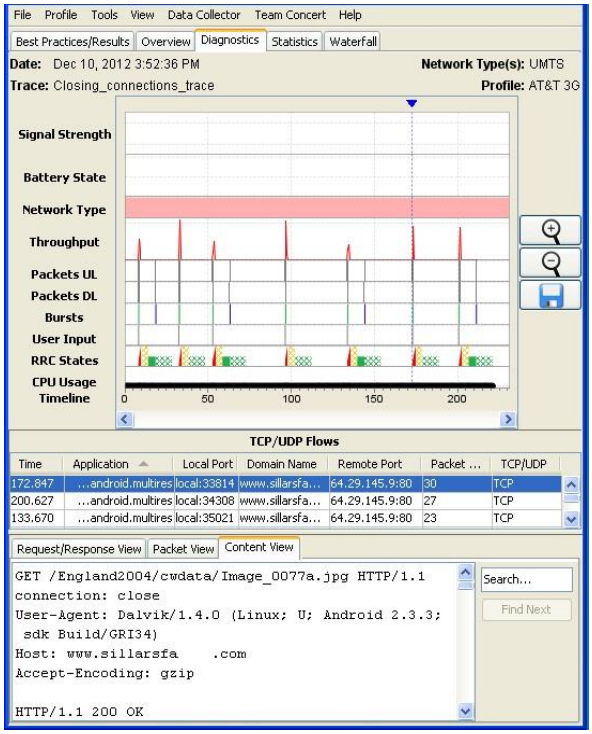
The Packet View tab shows information about the individual packets associated with the selected TCP or UDP flow (**Figure**).



The following table describes the columns of data in the Packet View tab.

| Column | Description |
|---|---|
| ID | An integer value that uniquely identifies each packet within the trace. |
| Time | The time, in seconds, from the beginning of the trace |
| Direction | The packet direction. One of the following values: UPLINK (The packet is sent up to the server), DOWNLINK (The packet is sent down from the server/host), UNKNOWN (The packet direction cannot be determined). |
| Type | Indicates the type of packet. One of the following values: OPEN_CONN (A packet that opens a connection), ACK (An acknowledgement packet), DATA (A data packet). |
| Payload Length | The length of the payload (the data being sent in the packet) in bytes. |
| **Column** | **Description** |
| TCP Flags | Each letter in this field represents a different TCP flag associated with the packet. More than one flag can be associated with a packet. The possible flags are: A - Ack; P - Push; R - Reset, S - Synchronize, F - Finish/End, E - Echo, U - Urgent, C - Congestion Windows Reduced. |

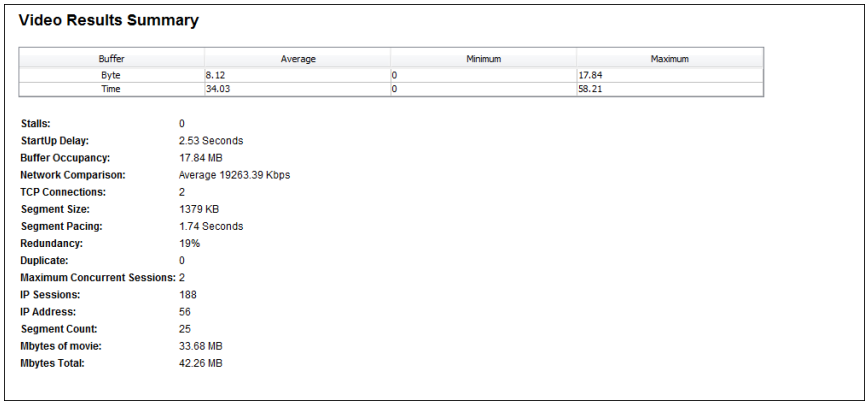
The Content View tab displays the content of the HTTP request/response (**Figure**). It has a Search field to the right of the content window, in which you can enter a search string. All instances of the string will be highlighted in the content window and the Find Next button can be used to navigate through the instances.




### 5.2.4. Video Tab

The Video tab breaks down the video usage during the data collection. This feature helps you understand how your application downloads video. If you have a trace with video collected, you will see this tab populated with Video Results Summary, Movie Manifest, and Video Requests.

A new table called Video Results Summary is available under the Video tab and it includes all the chosen movie related information such as bytes/seconds buffer, stalls, start-up delay, buffer occupancy, segments information, IP sessions/addresses, maximum concurrent sessions, TCP connections, redundancy etc. In this version of Video Optimizer, you can manage and analyze multiple manifest files in the same trace. The Start-up delay reminder pop up is available whenever you have video in the traffic for analysis.



**Video Results Summary**

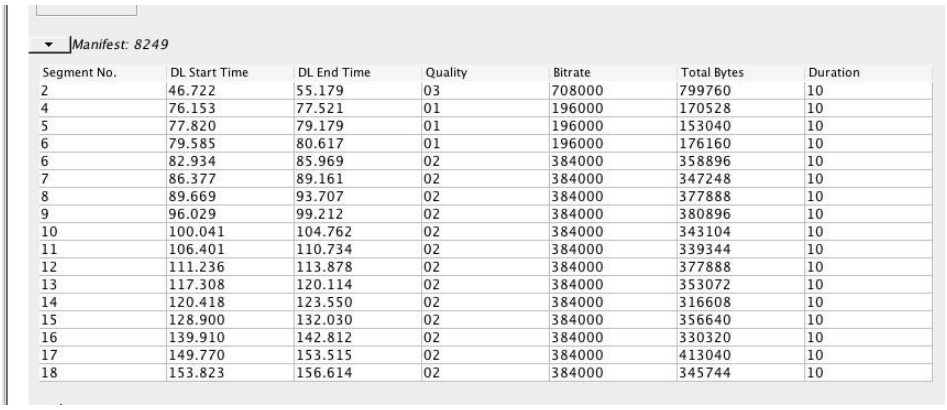
| Buffer | Average | Minimum | Maximum |
|---|---|---|---|
| Byte | 8.12 | 0 | 17.84 |
| Time | 34.03 | 0 | 58.21 |

| | |
|---|---|
| **Stalls:** | 0 |
| **StartUp Delay:** | 2.53 Seconds |
| **Buffer Occupancy:** | 17.84 MB |
| **Network Comparison:** | Average 19263.39 Kbps |
| **TCP Connections:** | 2 |
| **Segment Size:** | 1379 KB |
| **Segment Pacing:** | 1.74 Seconds |
| **Redundancy:** | 19% |
| **Duplicate:** | 0 |
| **Maximum Concurrent Sessions:** | 2 |
| **IP Sessions:** | 188 |
| **IP Address:** | 56 |
| **Segment Count:** | 25 |
| **Mbytes of movie:** | 33.68 MB |
| **Mbytes Total:** | 42.26 MB |


### *5.2.4.1. Movie Manifest Examples*

If Video Optimizer can read the movie manifest, it will break down the observed videos into tables.


Click the arrow  for one of the manifests to display a table with each segment that is observed from that manifest.



*Manifest: 8249*

| Segment No. | DL Start Time | DL End Time | Quality | Bitrate | Total Bytes | Duration |
|---|---|---|---|---|---|---|
| 2 | 46.722 | 55.179 | 03 | 708000 | 799760 | 10 |
| 4 | 76.153 | 77.521 | 01 | 196000 | 170528 | 10 |
| 5 | 77.820 | 79.179 | 01 | 196000 | 153040 | 10 |
| 6 | 79.585 | 80.617 | 01 | 196000 | 176160 | 10 |
| 6 | 82.934 | 85.969 | 02 | 384000 | 358896 | 10 |
| 7 | 86.377 | 89.161 | 02 | 384000 | 347248 | 10 |
| 8 | 89.669 | 93.707 | 02 | 384000 | 377888 | 10 |
| 9 | 96.029 | 99.212 | 02 | 384000 | 380896 | 10 |
| 10 | 100.041 | 104.762 | 02 | 384000 | 343104 | 10 |
| 11 | 106.401 | 110.734 | 02 | 384000 | 339344 | 10 |
| 12 | 111.236 | 113.878 | 02 | 384000 | 377888 | 10 |
| 13 | 117.308 | 120.114 | 02 | 384000 | 353072 | 10 |
| 14 | 120.418 | 123.550 | 02 | 384000 | 316608 | 10 |
| 15 | 128.900 | 132.030 | 02 | 384000 | 356640 | 10 |
| 16 | 139.910 | 142.812 | 02 | 384000 | 330320 | 10 |
| 17 | 149.770 | 153.515 | 02 | 384000 | 413040 | 10 |
| 18 | 153.823 | 156.614 | 02 | 384000 | 345744 | 10 |

**Figure** shows that segment number 2 and segments 4 through 18 were downloaded. Segment number 6 was downloaded once at Quality 01 with Bitrate 196k and a second time at Quality 02 with Bitrate 384k. The table also

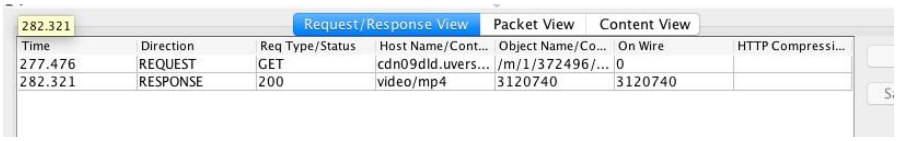
shows the time when each segment was downloaded, the number of bytes, and the duration of the video.

**Figure** shows a manifest with only two segments, both with number 113, and they were downloaded about one minute apart. Click either segment to view the Diagnostics tab.

Manifest: Whatever_It_Takes_30_1489375254_165772_113

| Segment No. | DL Start Time | DL End Time | Quality | Bitrate | Total Bytes | Duration |
|---|---|---|---|---|---|---|
| 113 | 282.321 | 301.255 | unknown | 0 | 3120740 | 0 |
| 113 | 329.100 | 358.049 | unknown | 0 | 3120740 | 0 |

The Diagnostics tab reveals that the segments from **Figure** are not actually movie segments, but an entire Mp4 video advertisement.



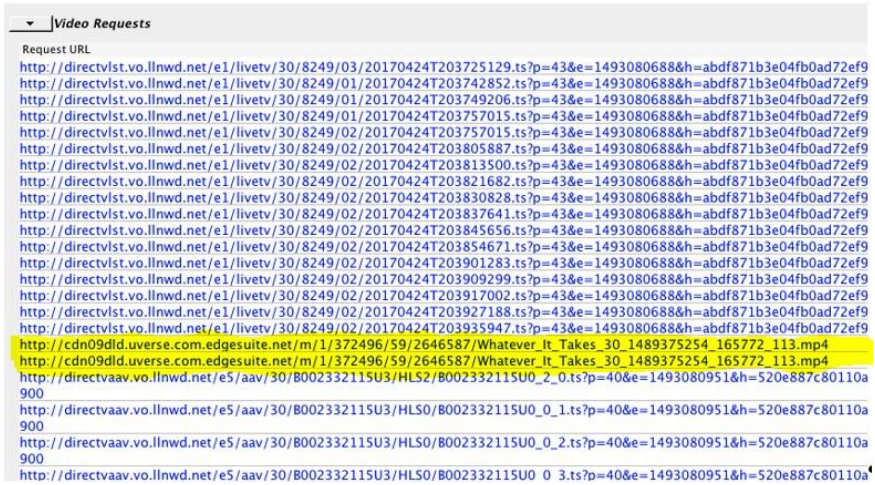
282.321 | Request/Response View | Packet View | Content View

| Time | Direction | Req Type/Status | Host Name/Cont... | Object Name/Co... | On Wire | HTTP Compressi... |
|---|---|---|---|---|---|---|
| 277.476 | REQUEST | GET | cdn09dld.uvers... | /m/1/372496/... | 0 | |
| 282.321 | RESPONSE | 200 | video/mp4 | 3120740 | 3120740 | |

This ad was downloaded twice during the trace (possibly due to slow network connectivity). A full 3 MB file that is downloaded twice can add significant delay to the end user (and it does appear as a failure in the duplicate file best practice). Video Optimizer uses the manifest and a URL parsing method to retrieve the values that are displayed in the Video tab.

### *5.2.4.2. Video Request Examples*

The last chart in the Video tab, Video Requests, is a list of all the URLs that are used in video requests:



Video Requests

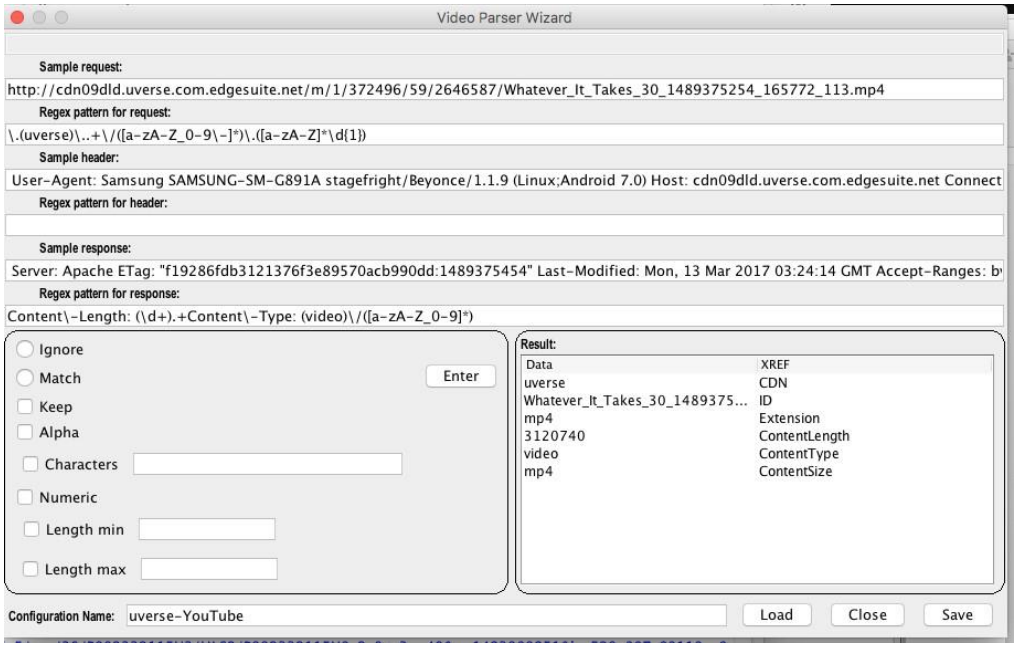
Request URL

http://directvlst.vo.llnwd.net/e1/livetv/30/8249/03/20170424T203725129.ts?p=43&e=1493080688&h=abdf871b3e04fb0ad72ef9
http://directvlst.vo.llnwd.net/e1/livetv/30/8249/01/20170424T203742852.ts?p=43&e=1493080688&h=abdf871b3e04fb0ad72ef9
http://directvlst.vo.llnwd.net/e1/livetv/30/8249/01/20170424T203749206.ts?p=43&e=1493080688&h=abdf871b3e04fb0ad72ef9
http://directvlst.vo.llnwd.net/e1/livetv/30/8249/01/20170424T203757015.ts?p=43&e=1493080688&h=abdf871b3e04fb0ad72ef9
http://directvlst.vo.llnwd.net/e1/livetv/30/8249/02/20170424T203757015.ts?p=43&e=1493080688&h=abdf871b3e04fb0ad72ef9
http://directvlst.vo.llnwd.net/e1/livetv/30/8249/02/20170424T203805887.ts?p=43&e=1493080688&h=abdf871b3e04fb0ad72ef9
http://directvlst.vo.llnwd.net/e1/livetv/30/8249/02/20170424T203813500.ts?p=43&e=1493080688&h=abdf871b3e04fb0ad72ef9
http://directvlst.vo.llnwd.net/e1/livetv/30/8249/02/20170424T203821682.ts?p=43&e=1493080688&h=abdf871b3e04fb0ad72ef9
http://directvlst.vo.llnwd.net/e1/livetv/30/8249/02/20170424T203830828.ts?p=43&e=1493080688&h=abdf871b3e04fb0ad72ef9
http://directvlst.vo.llnwd.net/e1/livetv/30/8249/02/20170424T203837641.ts?p=43&e=1493080688&h=abdf871b3e04fb0ad72ef9
http://directvlst.vo.llnwd.net/e1/livetv/30/8249/02/20170424T203845656.ts?p=43&e=1493080688&h=abdf871b3e04fb0ad72ef9
http://directvlst.vo.llnwd.net/e1/livetv/30/8249/02/20170424T203854671.ts?p=43&e=1493080688&h=abdf871b3e04fb0ad72ef9
http://directvlst.vo.llnwd.net/e1/livetv/30/8249/02/20170424T203901283.ts?p=43&e=1493080688&h=abdf871b3e04fb0ad72ef9
http://directvlst.vo.llnwd.net/e1/livetv/30/8249/02/20170424T203909299.ts?p=43&e=1493080688&h=abdf871b3e04fb0ad72ef9
http://directvlst.vo.llnwd.net/e1/livetv/30/8249/02/20170424T203917002.ts?p=43&e=1493080688&h=abdf871b3e04fb0ad72ef9
http://directvlst.vo.llnwd.net/e1/livetv/30/8249/02/20170424T203927188.ts?p=43&e=1493080688&h=abdf871b3e04fb0ad72ef9
http://directvlst.vo.llnwd.net/e1/livetv/30/8249/02/20170424T203935947.ts?p=43&e=1493080688&h=abdf871b3e04fb0ad72ef9
http://cdn09dld.uverse.com.edgesuite.net/m/1/372496/59/2646587/Whatever_It_Takes_30_1489375254_165772_113.mp4
http://cdn09dld.uverse.com.edgesuite.net/m/1/372496/59/2646587/Whatever_It_Takes_30_1489375254_165772_113.mp4
http://directvaav.vo.llnwd.net/e5/aav/30/B002332115U3/HLS2/B002332115U0_2_0.ts?p=40&e=1493080951&h=520e887c80110a900
http://directvaav.vo.llnwd.net/e5/aav/30/B002332115U3/HLS0/B002332115U0_0_1.ts?p=40&e=1493080951&h=520e887c80110a900
http://directvaav.vo.llnwd.net/e5/aav/30/B002332115U3/HLS0/B002332115U0_0_2.ts?p=40&e=1493080951&h=520e887c80110a900
http://directvaav.vo.llnwd.net/e5/aav/30/B002332115U3/HLS0/B002332115U0_0_3.ts?p=40&e=1493080951&h=520e887c80110a

Notice the two requests for the Whatever It Takes manifest.

Click one of these video requests to open the Video Parser Wizard, a tool to aid in the parsing of the video segments – for any video segment that might not be appearing properly in the manifest chart.

### *5.2.4.3. Video Parser Wizard*



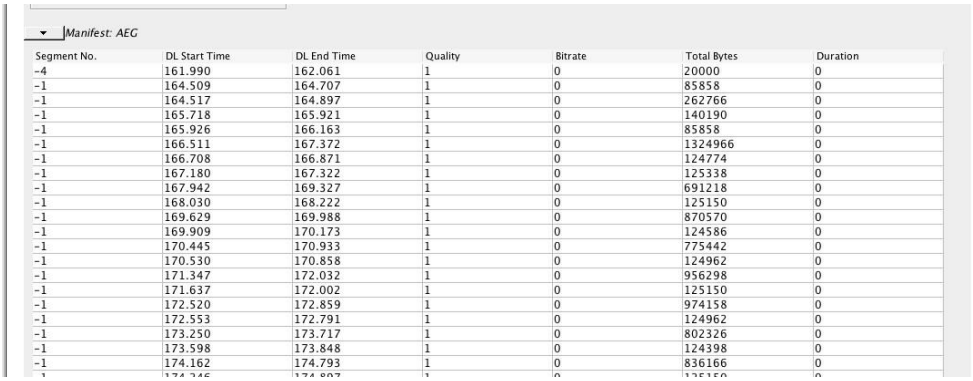

The Wizard uses regular expressions (regex) to find parameters in the URL, request and response headers in order to properly display each segment in the manifest chart.

In **Figure**, Video Optimizer is using a configuration called uverse-YouTube to dissect the segment parameters. It gets most of the information, although you could argue that the segment numbers are not totally accurate. Currently, Video Optimizer only requires the following parameters to perform video analysis:

- Video ID
- Segment number
- Quality
- Extension

One of the advantages of the Video Wizard is when you have a video type that has not yet been profiled. For

example, the trace in **Figure** shows every segment numbered as "-1".



Manifest: AEG

| Segment No. | DL Start Time | DL End Time | Quality | Bitrate | Total Bytes | Duration |
|---|---|---|---|---|---|---|
| -4 | 161.990 | 162.061 | 1 | 0 | 20000 | 0 |
| -1 | 164.509 | 164.707 | 1 | 0 | 85858 | 0 |
| -1 | 164.517 | 164.897 | 1 | 0 | 262766 | 0 |
| -1 | 165.718 | 165.921 | 1 | 0 | 140190 | 0 |
| -1 | 165.926 | 166.163 | 1 | 0 | 85858 | 0 |
| -1 | 166.511 | 167.372 | 1 | 0 | 1324966 | 0 |
| -1 | 166.708 | 166.871 | 1 | 0 | 124774 | 0 |
| -1 | 167.180 | 167.322 | 1 | 0 | 125338 | 0 |
| -1 | 167.942 | 169.327 | 1 | 0 | 691218 | 0 |
| -1 | 168.030 | 168.222 | 1 | 0 | 125150 | 0 |
| -1 | 169.629 | 169.988 | 1 | 0 | 870570 | 0 |
| -1 | 169.909 | 170.173 | 1 | 0 | 124586 | 0 |
| -1 | 170.445 | 170.933 | 1 | 0 | 775442 | 0 |
| -1 | 170.530 | 170.858 | 1 | 0 | 124962 | 0 |
| -1 | 171.347 | 172.032 | 1 | 0 | 956298 | 0 |
| -1 | 171.637 | 172.002 | 1 | 0 | 125150 | 0 |
| -1 | 172.520 | 172.859 | 1 | 0 | 974158 | 0 |
| -1 | 172.553 | 172.791 | 1 | 0 | 124962 | 0 |
| -1 | 173.250 | 173.717 | 1 | 0 | 802326 | 0 |
| -1 | 173.598 | 173.848 | 1 | 0 | 124398 | 0 |
| -1 | 174.162 | 174.793 | 1 | 0 | 836166 | 0 |

You can use the Video Parser Wizard to properly identify the parameters in the URL and the headers by building a regular expression to find them. For more information about using regular expressions, see the following websites.

https://en.wikipedia.org/wiki/Regular_expression

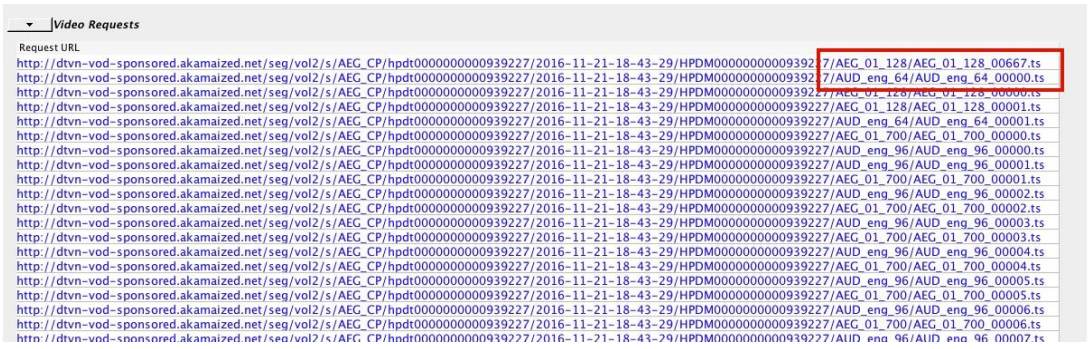
http://www.regular-expressions.info/

For our purposes, we will cover a few important features:

**Escaping:** Special characters require a backslash to escape them in a search. For example, if you want to denote a '/' found in a url, you would write it as '\/' to escape the character.

**Capture Groups:** These are the parameters that we want to "keep" for analysis. Capture groups are surrounded by parentheses. (this\ is\ being\ captured) will grab a string "this is being captured." Note the escaped spaces.

With that – let's dig into building the profile for this manifest. **Figure** shows the request table of a trace that is not properly profiled.



Video Requests

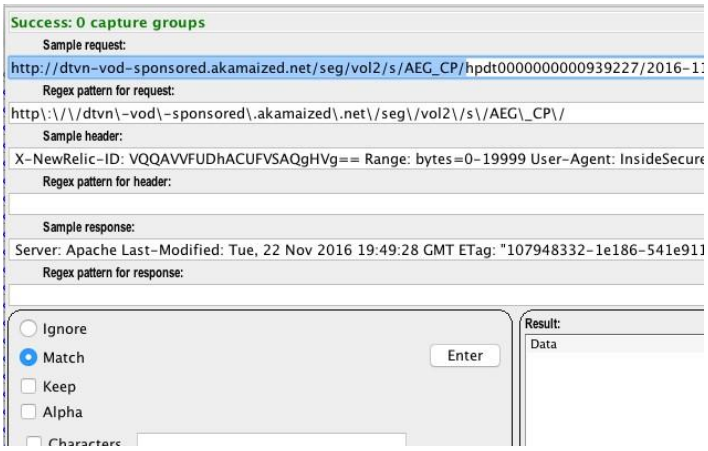
| Request URL |
|---|
| http://dtvn-vod-sponsored.akamaized.net/seg/vol2/s/AEG_CP/hpdt0000000000939227/2016-11-21-18-43-29/HPDM0000000000939227/AEG_01_128/AEG_01_128_00667.ts |
| http://dtvn-vod-sponsored.akamaized.net/seg/vol2/s/AEG_CP/hpdt0000000000939227/2016-11-21-18-43-29/HPDM0000000000939227/AUD_eng_64/AUD_eng_64_00000.ts |
| http://dtvn-vod-sponsored.akamaized.net/seg/vol2/s/AEG_CP/hpdt0000000000939227/2016-11-21-18-43-29/HPDM0000000000939227/AEG_01_128/AEG_01_128_00000.ts |
| http://dtvn-vod-sponsored.akamaized.net/seg/vol2/s/AEG_CP/hpdt0000000000939227/2016-11-21-18-43-29/HPDM0000000000939227/AEG_01_128/AEG_01_128_00001.ts |
| http://dtvn-vod-sponsored.akamaized.net/seg/vol2/s/AEG_CP/hpdt0000000000939227/2016-11-21-18-43-29/HPDM0000000000939227/AUD_eng_64/AUD_eng_64_00001.ts |
| http://dtvn-vod-sponsored.akamaized.net/seg/vol2/s/AEG_CP/hpdt0000000000939227/2016-11-21-18-43-29/HPDM0000000000939227/AEG_01_700/AEG_01_700_00000.ts |
| http://dtvn-vod-sponsored.akamaized.net/seg/vol2/s/AEG_CP/hpdt0000000000939227/2016-11-21-18-43-29/HPDM0000000000939227/AUD_eng_96/AUD_eng_96_00000.ts |
| http://dtvn-vod-sponsored.akamaized.net/seg/vol2/s/AEG_CP/hpdt0000000000939227/2016-11-21-18-43-29/HPDM0000000000939227/AUD_eng_96/AUD_eng_96_00001.ts |
| http://dtvn-vod-sponsored.akamaized.net/seg/vol2/s/AEG_CP/hpdt0000000000939227/2016-11-21-18-43-29/HPDM0000000000939227/AEG_01_700/AEG_01_700_00001.ts |
| http://dtvn-vod-sponsored.akamaized.net/seg/vol2/s/AEG_CP/hpdt0000000000939227/2016-11-21-18-43-29/HPDM0000000000939227/AUD_eng_96/AUD_eng_96_00002.ts |
| http://dtvn-vod-sponsored.akamaized.net/seg/vol2/s/AEG_CP/hpdt0000000000939227/2016-11-21-18-43-29/HPDM0000000000939227/AEG_01_700/AEG_01_700_00002.ts |
| http://dtvn-vod-sponsored.akamaized.net/seg/vol2/s/AEG_CP/hpdt0000000000939227/2016-11-21-18-43-29/HPDM0000000000939227/AUD_eng_96/AUD_eng_96_00003.ts |
| http://dtvn-vod-sponsored.akamaized.net/seg/vol2/s/AEG_CP/hpdt0000000000939227/2016-11-21-18-43-29/HPDM0000000000939227/AEG_01_700/AEG_01_700_00003.ts |
| http://dtvn-vod-sponsored.akamaized.net/seg/vol2/s/AEG_CP/hpdt0000000000939227/2016-11-21-18-43-29/HPDM0000000000939227/AUD_eng_96/AUD_eng_96_00004.ts |
| http://dtvn-vod-sponsored.akamaized.net/seg/vol2/s/AEG_CP/hpdt0000000000939227/2016-11-21-18-43-29/HPDM0000000000939227/AEG_01_700/AEG_01_700_00004.ts |
| http://dtvn-vod-sponsored.akamaized.net/seg/vol2/s/AEG_CP/hpdt0000000000939227/2016-11-21-18-43-29/HPDM0000000000939227/AUD_eng_96/AUD_eng_96_00005.ts |
| http://dtvn-vod-sponsored.akamaized.net/seg/vol2/s/AEG_CP/hpdt0000000000939227/2016-11-21-18-43-29/HPDM0000000000939227/AEG_01_700/AEG_01_700_00005.ts |
| http://dtvn-vod-sponsored.akamaized.net/seg/vol2/s/AEG_CP/hpdt0000000000939227/2016-11-21-18-43-29/HPDM0000000000939227/AUD_eng_96/AUD_eng_96_00006.ts |
| http://dtvn-vod-sponsored.akamaized.net/seg/vol2/s/AEG_CP/hpdt0000000000939227/2016-11-21-18-43-29/HPDM0000000000939227/AEG_01_700/AEG_01_700_00006.ts |
| http://dtvn-vod-sponsored.akamaized.net/seg/vol2/s/AEG_CP/hpdt0000000000939227/2016-11-21-18-43-29/HPDM0000000000939227/AUD_eng_96/AUD_eng_96_00007.ts |

The request table reveals that there might be two concurrent streams – one for video, and the other for audio, so you should create two profiles, starting with the “AEG” URL (below).

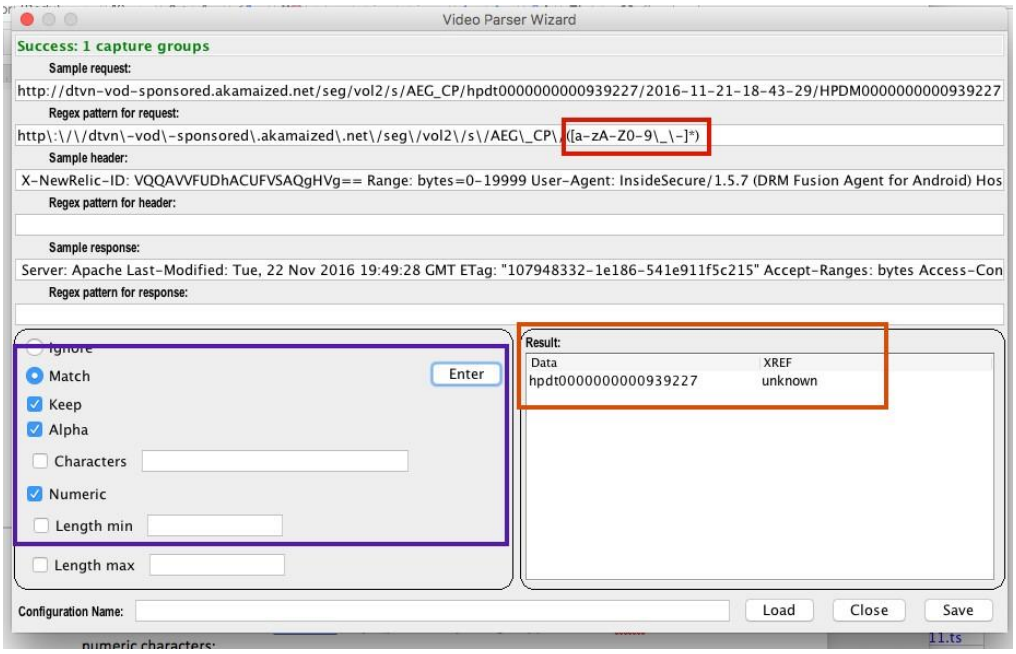
```
http://dtvn-vod-
sponsored.akamaized.net/seg/vol2/s/AEG_CP/hpdt0000000000939227/
201611-21-18-43-
29/HPDM0000000000939227/AEG_01_128/AEG_01_128_00667.ts
```

The beginning of the URL shows that it is coming from Akamai, and that it is from DTVnow, VOD “video on demand”. While this is useful information, skip it for now.

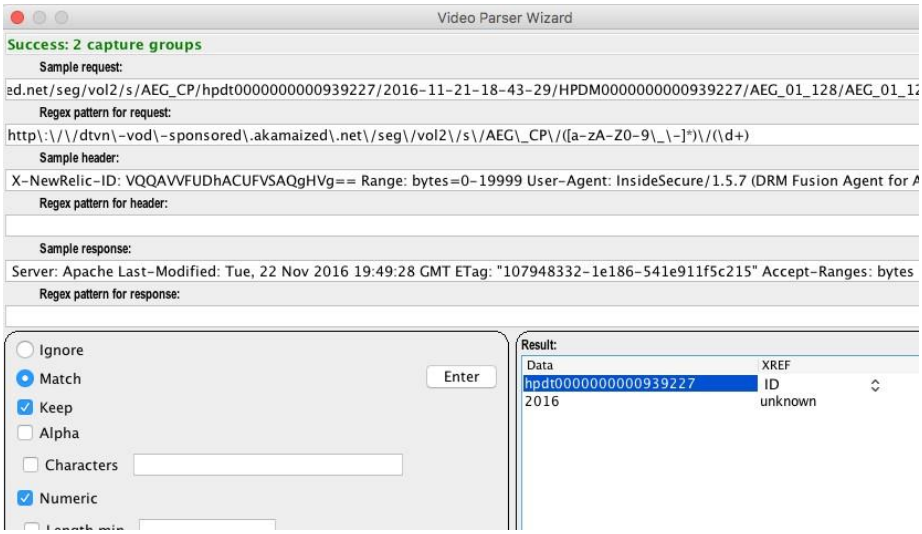
1. Highlight the URL, select Match, and click Enter to see a regular expression appear for that term. If you also select Keep, the Wizard will add the capture group (more on this later).



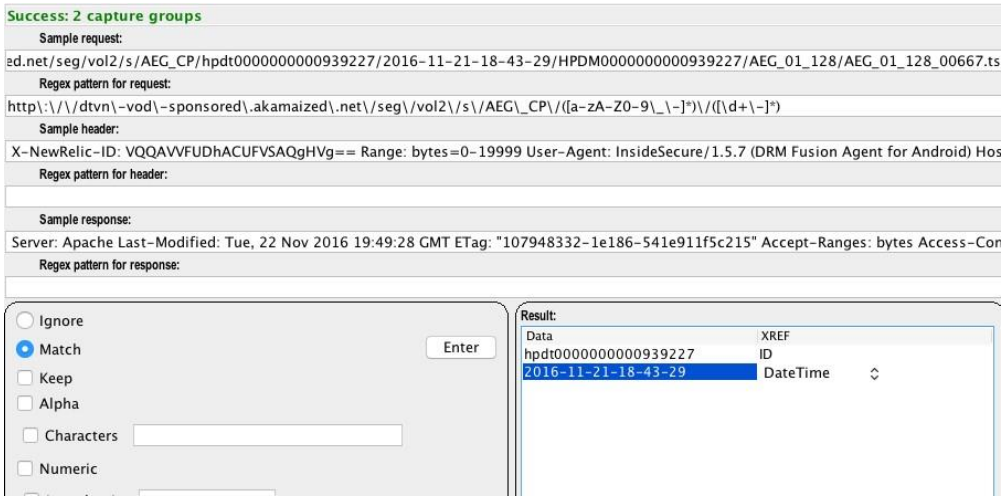

2. The next term in the URL is hpdt0000000000939227. This is probably the unique string for the video stream. Since it has both letters and numbers (and will change across videos, select Match, keep (put in a capture group), and Alpha and Numeric




3. Click Enter to see the new regular expression (red box in **Figure**) and because it is a capture group, there is a new string in the Result field (orange box). This means that you have successfully captured your first parameter.
4. Right-click on “unknown” in the Result field – so that you can apply and add the XREF – in this case “ID” might be a good choice for your XREF.
5. Notice the date in the URL (after the next /). Add “\/” to the regular expression which means, stop at the next forward slash.
6. To apply the date, add a search for numeric (d+), but this will only get to the first dash (the result only shows 2016).



7. Manually edit the added capture group from (d+) to ([\d+\-]*) to look for digits or dashes, and the correct string is populated (**Figure**).

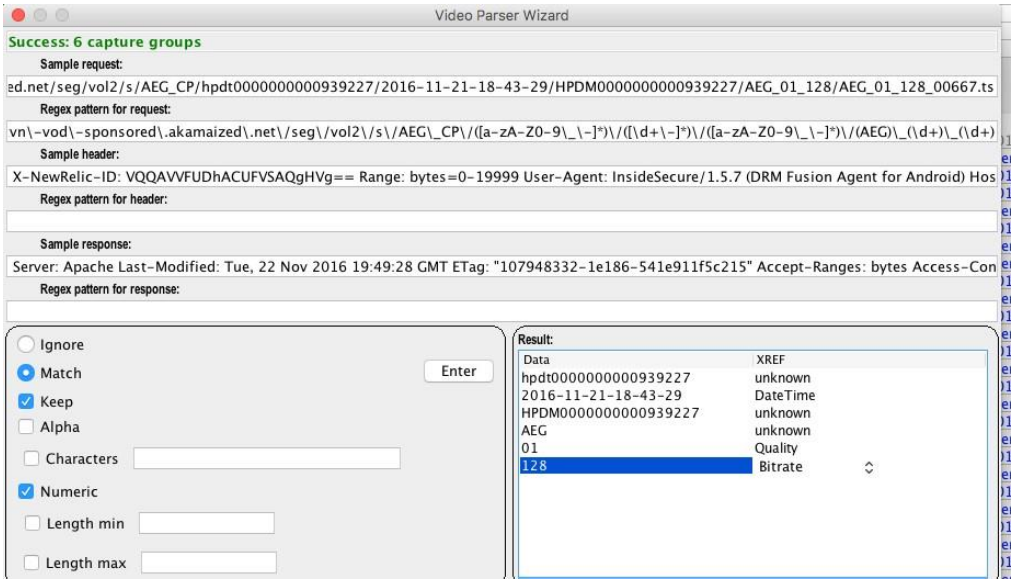



After the next forward slash, there is another instance of the ID (with capital letters this time), so add "\/([a-zA-Z0-9_\-]*)\/" to the regex to find this string.

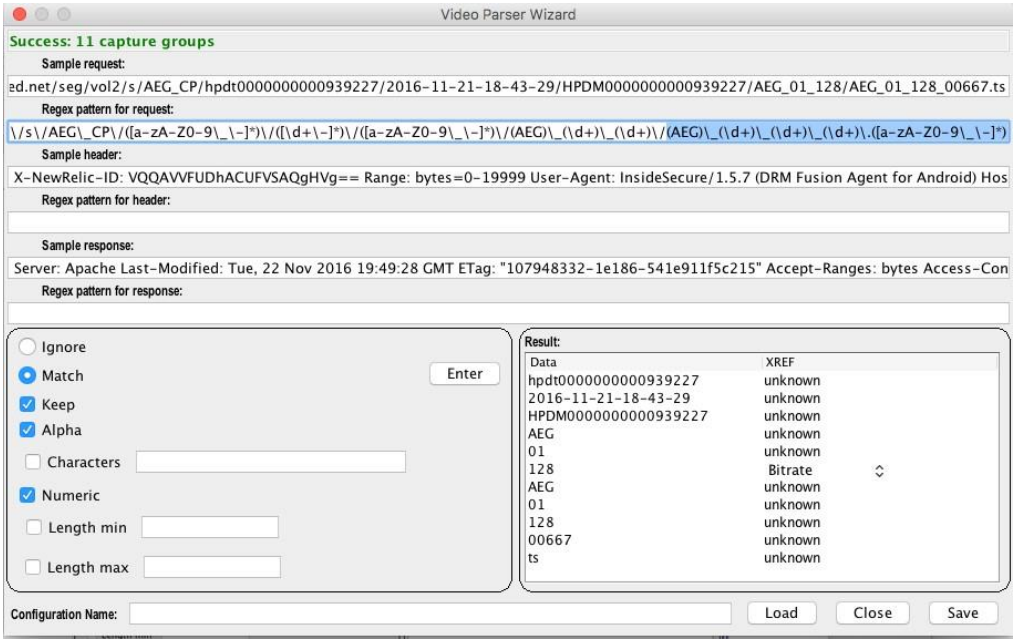
Now you come to the part where the two URL types are different (AEG and AUD)—and you must ensure that the regular expression you create will only work for the URL in question.

***To find the AEG_01_128 files:***

1. Match & Keep on the String AEG.
2. Add _ to pass the underscore.
3. Add Keep and Numeric to find 01.
4. Add _ to pass the underscore.
5. Add Keep and Numeric for the 128.
6. The regular expression appears as (AEG)_(\d+)_(\d+) 7. Apply Quality to 128, and Bitrate to 128.




After the next “/”, the same terms reappear, followed by the segment number, a period and the extension. The regex looks like the following: (AEG)_(\d+)_(\d+)_(\d+)\.([a-zA-Z0-9_\-]*) The final view of the Wizard is displayed:

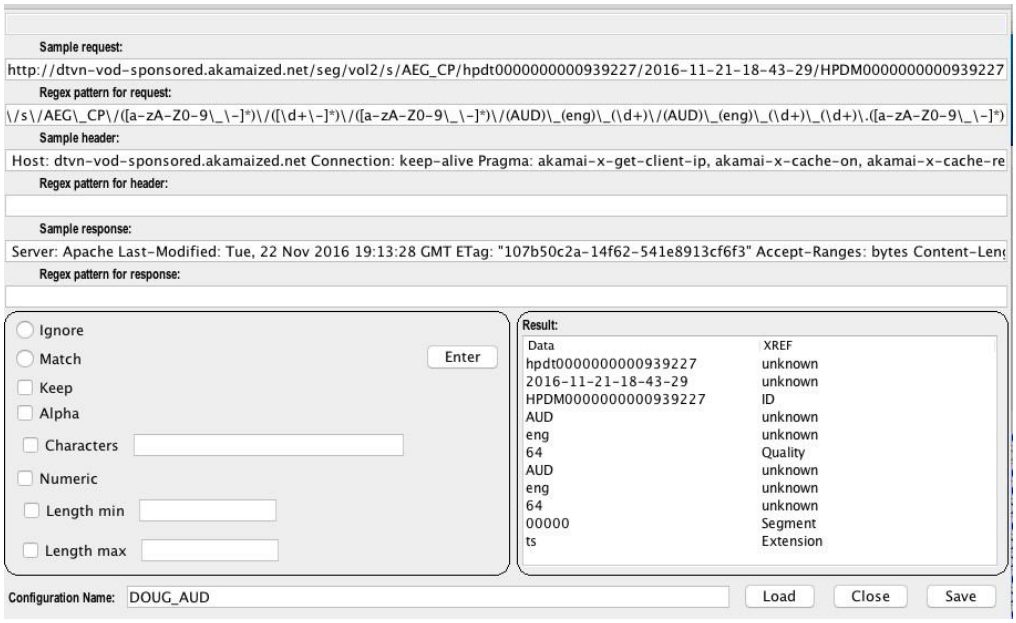



8. Save this file, and then reload the trace to apply this JSON profile to the video.

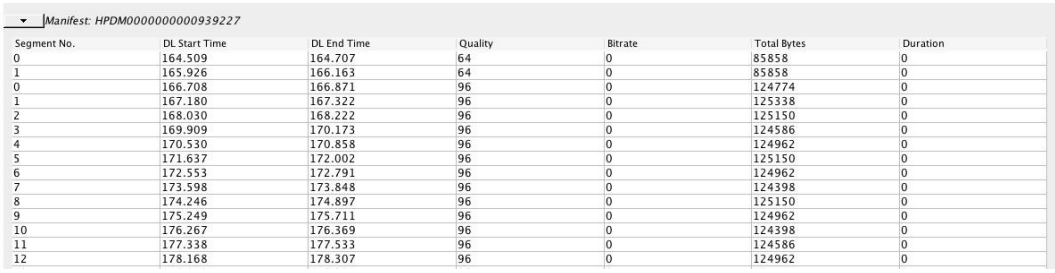
***To find the AUD files:***

1. Modify the same regex to find the AUD files (All AEG -> AUD, and the first (/d+) to (eng) were the only changes), for example:

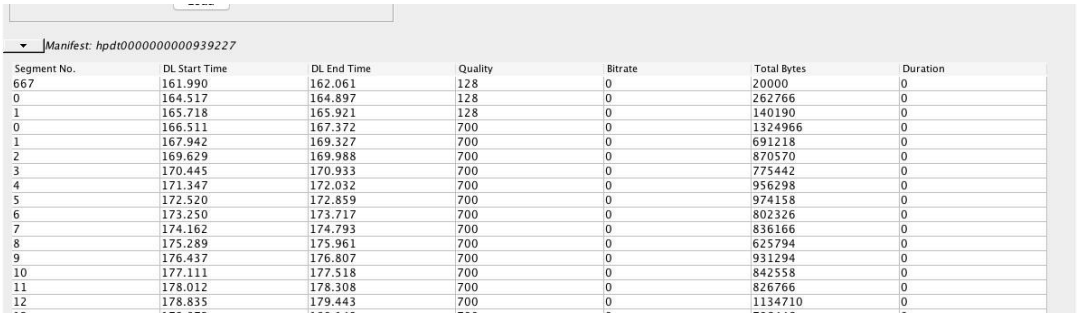
   http\:\/\/dtvn\-vod\-sponsored\.akamaized\.net\/seg\/vol2\/s\/AEG_CP\/([a-zA-Z09_\-]*)\/([\d+\-]*)\/([a-zA-Z0-9_\-]*)\/(AUD)_(eng)_(\d+)\/(AUD)_(eng)_(\d+)_(\d+)\.([a-zA-Z0-9_\-]*)




2. Reload the trace to see two manifest files, one for the audio tracks, and the other with the video tracks:




| Segment No. | DL Start Time | DL End Time | Quality | Bitrate | Total Bytes | Duration |
|---|---|---|---|---|---|---|
| 0 | 164.509 | 164.707 | 64 | 0 | 85858 | 0 |
| 1 | 165.926 | 166.163 | 64 | 0 | 85858 | 0 |
| 0 | 166.708 | 166.871 | 96 | 0 | 124774 | 0 |
| 1 | 167.180 | 167.322 | 96 | 0 | 125338 | 0 |
| 2 | 168.030 | 168.222 | 96 | 0 | 125150 | 0 |
| 3 | 169.909 | 170.173 | 96 | 0 | 124586 | 0 |
| 4 | 170.530 | 170.858 | 96 | 0 | 124962 | 0 |
| 5 | 171.637 | 172.002 | 96 | 0 | 125150 | 0 |
| 6 | 172.553 | 172.791 | 96 | 0 | 124962 | 0 |
| 7 | 173.598 | 173.848 | 96 | 0 | 124398 | 0 |
| 8 | 174.246 | 174.897 | 96 | 0 | 125150 | 0 |
| 9 | 175.249 | 175.711 | 96 | 0 | 124962 | 0 |
| 10 | 176.267 | 176.369 | 96 | 0 | 124398 | 0 |
| 11 | 177.338 | 177.533 | 96 | 0 | 124586 | 0 |
| 12 | 178.168 | 178.307 | 96 | 0 | 124962 | 0 |



Manifest: hpdt0000000000939227

| Segment No. | DL Start Time | DL End Time | Quality | Bitrate | Total Bytes | Duration |
|---|---|---|---|---|---|---|
| 667 | 161.990 | 162.061 | 128 | 0 | 20000 | 0 |
| 0 | 164.517 | 164.897 | 128 | 0 | 262766 | 0 |
| 1 | 165.718 | 165.921 | 128 | 0 | 140190 | 0 |
| 0 | 166.511 | 167.372 | 700 | 0 | 1324966 | 0 |
| 1 | 167.942 | 169.327 | 700 | 0 | 691218 | 0 |
| 2 | 169.629 | 169.988 | 700 | 0 | 870570 | 0 |
| 3 | 170.445 | 170.933 | 700 | 0 | 775442 | 0 |
| 4 | 171.347 | 172.032 | 700 | 0 | 956298 | 0 |
| 5 | 172.520 | 172.859 | 700 | 0 | 974158 | 0 |
| 6 | 173.250 | 173.717 | 700 | 0 | 802326 | 0 |
| 7 | 174.162 | 174.793 | 700 | 0 | 836166 | 0 |
| 8 | 175.289 | 175.961 | 700 | 0 | 625794 | 0 |
| 9 | 176.437 | 176.807 | 700 | 0 | 931294 | 0 |
| 10 | 177.111 | 177.518 | 700 | 0 | 842558 | 0 |
| 11 | 178.012 | 178.308 | 700 | 0 | 826766 | 0 |
| 12 | 178.835 | 179.443 | 700 | 0 | 1134710 | 0 |

These two scripts will now parse any similar movie segments in future traces of this mobile application.

### 5.2.5. Statistics Tab

The Statistics tab displays key statistical information based on the analysis of the loaded trace. The information on the Statistics Tab is divided into the following sections:

- Header
- TCP(Session) Statistics
- Endpoint Summary
- RRC(Radio Resource Control) State Machine Simulation
- Burst Analysis
- HTTP Cache Statistics
- Energy Efficiency Simulation

#### *5.2.5.1. Export Button*

The top right corner of the Statistics tab has an Export button.


1. Click Export.
2. Click Save in the Save As dialog box to save all of the data from the Diagnostics tab and the Statistics tab as a *.csv* file.
3. Click Close when the file has been saved or click Open to open the .csv file immediately. When you click Open, the export file will be opened using the program that you have identified in your OS as the default program for .csv files.

### *5.2.5.2. Header Section*

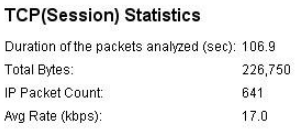
The Header section of the Statistics tab displays information about the trace that the statistics are derived from. The information in the Header section is the same as in the header of the Best Practices/Results tab.

The following table describes the information displayed in the Header section:

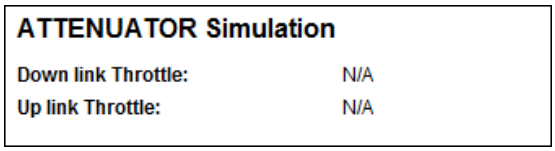
| Label | Description |
|---|---|
| Date | The date when the trace files were generated. |
| Trace | The name of the folder containing the trace files. |
| Application(s) Name: Version | The names and versions of the applications that were running when the trace data was collected. |
| **Label** | **Description** |
| Data Collector Version | The version of the Video Optimizer Data Collector that was used to collect the trace data. |
| Device make/model | The make and model of the device from which the data was collected. |
| OS/Platform Version | The operating system version or platform version of the device that the trace was captured on. |
| Network type | The type of network, like 3G or LTE, which was in use when the data was collected. |
| Profile | The device profile that was used for the trace analysis. |

### *5.2.5.3. TCP (Session) Statistics*

The TCP (Session) Statistics section of the Statistics Tab information page provides overall statistics about the TCP Packet information captured in the loaded trace files.



#### *5.2.5.4. ATTENUATOR Simulation*




The following table describes the statistics contained in the TCP (Session) Statistics section:

| Field | Description |
|---|---|
| Duration of the packets analyzed (sec) | The time difference, in seconds, between the last packet time stamp and the first packet time stamp in the loaded trace. |
| Total Bytes | The sum of the packet length values from the loaded trace. The packet length value includes both the header length and the data length. |
| IP Packet Count | The total number of packets in the loaded trace. |
| Avg Rate (kbps) | The average transfer rate of data in kilobytes per second. This value is derived from the total number of transferred bytes and the trace duration. |

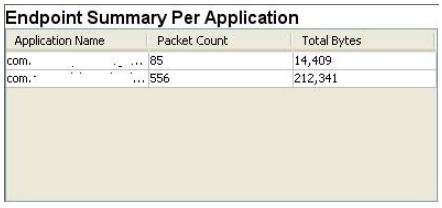
#### *5.2.5.5. Endpoint Summary Section*

The Endpoint Summary section of the Statistics Tab information page contains two tables that summarize the packet information for each application, and for each IP Address. These two tables are:

- Endpoint Summary Per Application
- Endpoint Summary Per IP Address

#### *5.2.5.6. Endpoint Summary Per Application Table*

The Endpoint Summary Per Application table summarizes the number of packets and total number of bytes for each application in the trace.

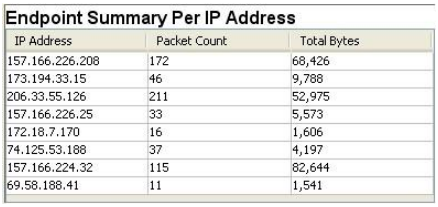



The following table describes the statistics contained in the Endpoint Summary Per Application table:

| Column | Description |
|---|---|
| Application Name | The name of the application. |
| Packet Count | The total number of packets for this application. |
| Total Bytes | The total number of bytes for this application. |

### *5.2.5.7. Endpoint Summary Per IP Address Table*

The Endpoint Summary Per IP Address table summarizes the number of packets and total number of bytes for each IP address in the trace.



Endpoint Summary Per IP Address

| IP Address | Packet Count | Total Bytes |
|---|---|---|
| 157.166.226.208 | 172 | 68,426 |
| 173.194.33.15 | 46 | 9,788 |
| 206.33.55.126 | 211 | 52,975 |
| 157.166.226.25 | 33 | 5,573 |
| 172.18.7.170 | 16 | 1,606 |
| 74.125.53.188 | 37 | 4,197 |
| 157.166.224.32 | 115 | 82,644 |
| 69.58.188.41 | 11 | 1,541 |

The following table describes the statistics contained in the Endpoint Summary Per IP Address table:

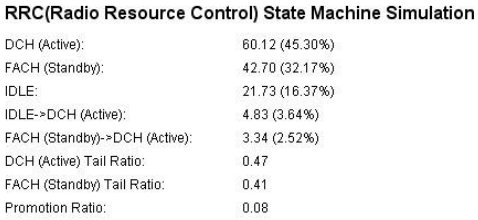
| Column | Description |
|---|---|
| IP Address | The IP Address. |
| Packet Count | The total number of packets for this IP address. |
| Total Bytes | The total number of bytes for this IP address. |

### *5.2.5.8. RRC (Radio Resource Control) State Machine Simulation*

The RRC (Radio Resource Control) State Machine Simulation section of the Statistics Tab information page displays an analysis of how much time was spent in the various RRC states.

**Note:** The names of the RRC states and the information displayed in this section depend on the type of device profile that is selected (3G, LTE, or WiFi).

When a 3G device profile like *AT&T 3G* is selected, then the section appears like the following:



**RRC(Radio Resource Control) State Machine Simulation**

| | |
|---|---|
| DCH (Active): | 60.12 (45.30%) |
| FACH (Standby): | 42.70 (32.17%) |
| IDLE: | 21.73 (16.37%) |
| IDLE->DCH (Active): | 4.83 (3.64%) |
| FACH (Standby)->DCH (Active): | 3.34 (2.52%) |
| DCH (Active) Tail Ratio: | 0.47 |
| FACH (Standby) Tail Ratio: | 0.41 |
| Promotion Ratio: | 0.08 |

The following table describes the statistics contained in the RRC (Radio Resource Control) State Machine Simulation section when a 3G device profile like *AT&T 3G* is selected:

| Field | Description |
|---|---|
| DCH (Active) | The amount of DCH (Active) state time, in seconds, and its percentage of total packet duration. |
| FACH (Standby) | The amount of FACH (Standby) state time, in seconds, and its percentage of total packet duration. |
| IDLE | The amount of IDLE state time, in seconds, and its percentage of total packet duration. |
| IDLE→DCH (Active) | The amount of time spent in promotion from the IDLE state to the DCH (Active) state, in seconds, and its percentage of total packet duration. |
| FACH (Standby) → DCH (Active) | The amount of time spent in promotion from the FACH (Standby) state to the DCH (Active) state, in seconds, and its percentage of total packet duration. |
| DCH (Active) Tail Ratio | The ratio between the amount of DCH (Active) Tail state time and the amount of DCH (Active) state time. |
| FACH (Standby) Tail Ratio | The ratio between the amount of FACH (Standby) Tail state time and the amount of FACH (Standby) state time. |
| Promotion Ratio | The ratio between the sums of the total promoted RRC states time and the total packet duration. The promoted RRC states are IDLE→DCH (Active) and FACH (Standby) →DCH (Active). |

When an LTE device profile like *AT&T LTE* is selected, then the section appears like the following:

**RRC(Radio Resource Control) State Machine Simulation**

| | |
|---|---|
| IDLE->Continuous Reception: | 0.78 (0.59% of time) |
| Continuous Reception: | 16.07 (12.11% of time) |
| Continuous Reception Tail: | 11.30 (8.51% of time) |
| Short DRX: | 2.17 (1.64% of time) |
| Long DRX: | 89.70 (67.58% of time) |
| IDLE: | 24.00 (18.08% of time) |
| Continuous Reception Tail Ratio: | 0.70 |
| Long DRX Ratio: | 0.02 |
| Short DRX Ratio: | 0.84 |
| Promotion Ratio: | 0.01 |

The following table describes the statistics contained in the RRC (Radio Resource Control) State Machine Simulation section when an LTE device profile is selected:

| Field | Description |
|---|---|
| IDLE->Continuous Reception | The amount of time spent in promotion from the IDLE state to Continuous Reception, in seconds, and its percentage of total packet duration. |
| Continuous Reception | The amount of Continuous Reception state time, in seconds, and its percentage of total packet duration. |
| Continuous Reception Tail | The amount of Continuous Reception Tail state time, in seconds, and its percentage of total packet duration. |
| Short DRX | The amount of Short DRX state time, in seconds, and its percentage of total packet duration. |
| Long DRX | The amount of Long DRX state time, in seconds, and its percentage of total packet duration. |
| IDLE | The amount of IDLE state time, in seconds, and its percentage of total packet duration. |
| Continuous Reception Tail Ratio | The ratio between the amount of Continuous Reception Tail state time and the amount of Continuous Reception state time. |
| Long DRX Ratio | The ratio between the amount of Long DRX state time and the amount of Continuous Reception and Short DRX state time. |
| Short DRX Ratio | The ratio between the amount of Short DRX state time and the amount of Continuous Reception and Long DRX state time. |
| Promotion Ratio | The ratio between the promoted state time (IDLE→Continuous Reception), and the sum of the IDLE, IDLE→Continuous Reception, Continuous Reception, and Continuous Reception Tail state times. |

When a WiFi device profile like *AT&T WiFi* is selected, then the section appears like the following:

**RRC(Radio Resource Control) State Machine Simulation**

| | |
|---|---|
| WiFi Active: | 28.36 (21.37% of time) |
| WiFi Tail: | 13.00 (9.80% of time) |
| WiFi Idle: | 104.36 (78.63% of time) |

The following table describes the statistics contained in the RRC (Radio Resource Control) State Machine Simulation section when a WiFi device profile is selected:

| Field | Description |
|---|---|
| WiFi Active | The amount of WiFi Active state time, in seconds, and its percentage of total packet duration. |
| WiFi Tail | The amount of WiFi Tail state time, in seconds, and its percentage of total packet duration. |
| WiFi Idle | The amount of WiFi Idle state time, in seconds, and its percentage of total packet duration. |

### *5.2.5.9. Burst Analysis Section*

The Burst Analysis section of the Statistics Tab information page, contains two tables that provide burst information. One that groups the bursts by burst type, and another that lists individual bursts. These two tables are:

- Burst Analysis
- Individual Burst Analysis

### *5.2.5.10. Burst Analysis Table*

The Burst Analysis table provides information about the collected bursts from the loaded trace, summarized by burst type. You can export the contents of this table in the CSV format by right-clicking on it.

**Note:** The columns of information displayed in this table depend on the type of device profile that is selected.

When a 3G device profile like *AT&T 3G* is selected, the Burst Analysis table appears like the following:

**Burst Analysis**

| Burst | Bytes | % of Bytes | Energy | % of Energy | DCH (Active) | % DCH (Active) | JpKB |
|---|---|---|---|---|---|---|---|
| TcpControl | 0 | 0.0 | 9.97 | 10.3 | 3.850 | 6.4 | 0.000 |
| UserInput | 5,968 | 3.1 | 27.38 | 28.3 | 15.693 | 26.1 | 0.574 |
| App | 182,733 | 94.5 | 45.01 | 46.5 | 31.196 | 51.9 | 0.031 |
| SvrNetDelay | 4,625 | 2.4 | 14.41 | 14.9 | 9.377 | 15.6 | 0.389 |

When an LTE device profile like *AT&T LTE* is selected, the Burst Analysis table appears like the following:

**Burst Analysis**

| Burst | Bytes | % of Bytes | Energy | % of Energy | Continuous Reception | % of Continuous Reception | JpKB |
|---|---|---|---|---|---|---|---|
| TcpControl | 0 | 0.0 | 21.29 | 17.6 | 0.810 | 5.0 | 0.000 |
| UserInput | 4,602 | 2.4 | 32.21 | 26.6 | 1.278 | 8.0 | 0.875 |
| App | 184,099 | 95.2 | 50.62 | 41.8 | 12.679 | 78.9 | 0.034 |
| SvrNetDelay | 4,625 | 2.4 | 17.08 | 14.1 | 1.301 | 8.1 | 0.462 |

When a WiFi device profile like *AT&T WiFi* is selected, the Burst Analysis table appears like the following:

**Burst Analysis**

| Burst | Bytes | % of Bytes | Energy | % of Energy | WiFi Active | % of WiFi Active | JpKB |
|---|---|---|---|---|---|---|---|
| TcpControl | 0 | 0.0 | 1.12 | 8.3 | 1.560 | 5.5 | 0.000 |
| UserInput | 4,602 | 2.4 | 1.75 | 13.0 | 2.963 | 10.4 | 0.048 |
| App | 184,099 | 95.2 | 9.24 | 68.3 | 21.042 | 74.2 | 0.006 |
| SvrNetDelay | 4,625 | 2.4 | 1.41 | 10.4 | 2.794 | 9.9 | 0.038 |

The following table describes all the statistics contained in the Burst Analysis table for all types of device profiles:

| Field | Description |
|---|---|
| Burst | One of the following Burst types according to the request/response types in the loaded trace. |

| Burst Categories |
|---|
| TCP Control |
| TCP Loss Recover |
| User Input |
| Screen Rotation |
| App |
| SvrNetDelay |
| NonTarget |
| LargeBurst |
| Periodical |
| Unknown |
| Userdef 1 |
| Userdef 2 |
| Userdef 3 |

| | |
|---|---|
| Bytes | The payload length, in bytes, for the corresponding Burst type. The payload length considers only the data length of packets which occurred during the burst. |

| Field | Description |
|---|---|
| % of Bytes | The percentage of total payload used by the individual burst payload. The total payload is the sum of all burst payloads. |
| Energy | The amount of Energy, in Joules, for the corresponding Burst type. |
| % of Energy | The percentage of total burst energy used by the individual burst. Total burst energy is the sum of all individual burst Energy amounts. |
| DCH (Active) | The amount of DCH Active time for the corresponding Burst type. Note: This column is only displayed when a 3G device profile is selected. |
| %DCH (Active) | The percentage of total DCH Time used by the individual burst. The total DCH Time is the sum of all individual burst DCH Times.<br><br>Note: This column is only displayed when a 3G device profile is selected. |
| Continuous Reception | The amount of Continuous Reception time for |

| | |
|---|---|
| | the corresponding Burst type.<br><br>Note: This column is only displayed when an LTE device profile is selected. |
| % of Continuous Reception | The percentage of total Continuous Reception time used by the individual burst. The total Continuous Reception time is the sum of all individual burst Continuous Reception times.<br><br>Note: This column is only displayed when an LTE device profile is selected. |
| WiFi Active | The amount of WiFi Active time for the corresponding Burst type.<br><br>Note: This column is only displayed when a WiFi device profile is selected. |
| % of WiFi Active | The percentage of total WiFi Active time used by the individual burst. The total WiFi Active time is the sum of all individual burst WiFi Active times.<br><br>Note: This column is only displayed when a WiFi device profile is selected. |
| JpKB | The number of Joules per Kilobytes for the |

| | |
|---|---|
| | corresponding Burst type calculated from the amount of burst type energy and burst type payload. |

### *5.2.5.11. Individual Burst Analysis Table*

The Individual Burst Analysis table provides information about each individual burst in the loaded trace. You can export the contents of this table in the CSV format by right-clicking on it.

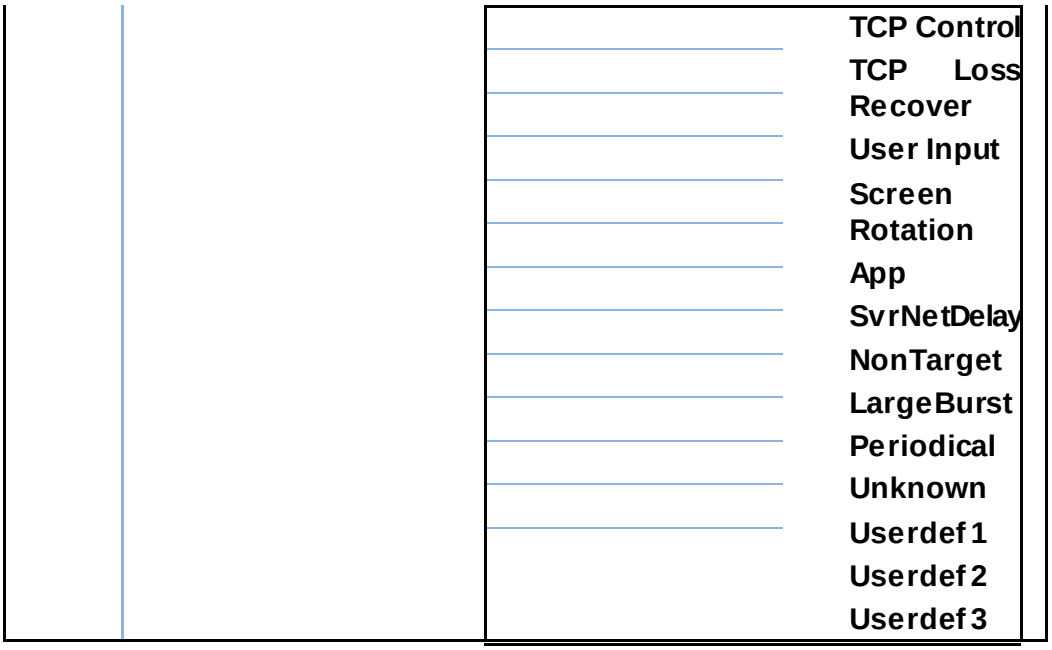
**Individual Burst Analysis**

| Start Time | Time Elapsed | Bytes | Packet Count | Burst |
|---|---|---|---|---|
| 14.465 | 5.240 | 78,568 | 146 | App |
| 23.757 | 12.496 | 42,367 | 171 | App |
| 38.155 | 3.138 | 24,380 | 91 | App |
| 43.477 | 7.985 | 37,418 | 124 | App |
| 55.323 | 1.091 | 0 | 32 | TcpControl |
| 60.104 | 0.656 | 1,229 | 10 | UserInput |
| 72.772 | 2.193 | 1,041 | 9 | UserInput |
| 85.046 | 0.037 | 0 | 2 | TcpControl |

The following table describes the statistics contained in the Individual Burst Analysis section:

| Field | Description |
|---|---|
| Start Time | The start time of the burst, in seconds, from the beginning of the trace. |
| Time Elapsed | The time elapsed during the burst, in seconds. |
| Bytes | The payload length, in bytes, for the burst. The payload length considers only the data length of packets which occurred during the burst. |
| Packet Count | The number of packets in the burst. |
| Burst | One of the following Burst types according to the request/response types in the loaded trace. |

**Burst Categories.**



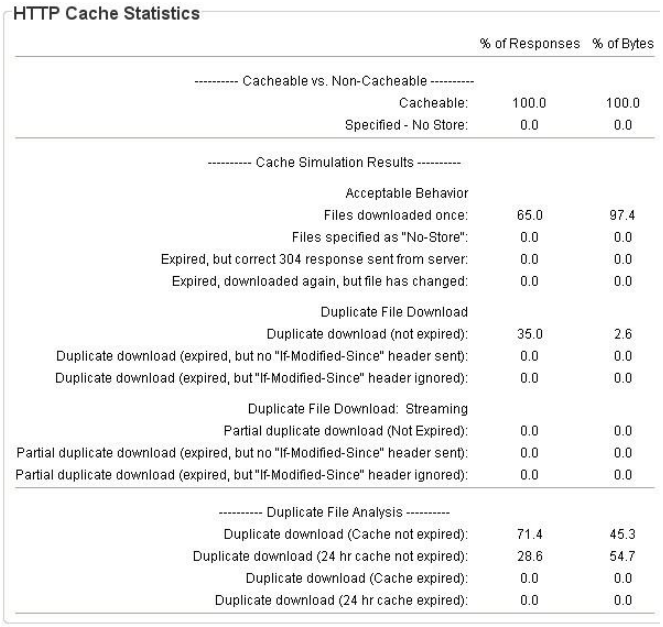

### *5.2.5.12. HTTP Cache Statistics*

The HTTP Cache Statistics section of the Statistics Tab information page displays statistical information about the cache based on the data in the loaded trace. Caching is the process of storing data on the client side to avoid the repeated download of data from the server. This increases the amount of bandwidth available for common requests and responses.

This section contains the following sub-categories:

- Cacheable vs. Non-Cacheable
- Cache Simulation Results
- Duplicate File Analysis

The following figure shows the columns and sub-categories of the HTTP Cache Statistics section.



HTTP Cache Statistics

| | % of Responses | % of Bytes |
|---|---|---|
| --------- Cacheable vs. Non-Cacheable --------- | | |
| Cacheable: | 100.0 | 100.0 |
| Specified - No Store: | 0.0 | 0.0 |
| --------- Cache Simulation Results --------- | | |
| Acceptable Behavior | | |
| Files downloaded once: | 65.0 | 97.4 |
| Files specified as "No-Store": | 0.0 | 0.0 |
| Expired, but correct 304 response sent from server: | 0.0 | 0.0 |
| Expired, downloaded again, but file has changed: | 0.0 | 0.0 |
| Duplicate File Download | | |
| Duplicate download (not expired): | 35.0 | 2.6 |
| Duplicate download (expired, but no "If-Modified-Since" header sent): | 0.0 | 0.0 |
| Duplicate download (expired, but "If-Modified-Since" header ignored): | 0.0 | 0.0 |
| Duplicate File Download: Streaming | | |
| Partial duplicate download (Not Expired): | 0.0 | 0.0 |
| Partial duplicate download (expired, but no "If-Modified-Since" header sent): | 0.0 | 0.0 |
| Partial duplicate download (expired, but "If-Modified-Since" header ignored): | 0.0 | 0.0 |
| --------- Duplicate File Analysis --------- | | |
| Duplicate download (Cache not expired): | 71.4 | 45.3 |
| Duplicate download (24 hr cache not expired): | 28.6 | 54.7 |
| Duplicate download (Cache expired): | 0.0 | 0.0 |
| Duplicate download (24 hr cache expired): | 0.0 | 0.0 |

The HTTP Cache Statistics section contains the following columns:

| Column | Description |
|---|---|
| % of Response | Displays the amount of responses for this row item expressed as a percentage of the total number of responses. |
| % of Bytes | Displays the number of bytes for this row item expressed as a percentage of the total number of bytes. |

### *5.2.5.13. Cacheable vs. Non-Cacheable*

The Cacheable vs. Non-Cacheable section of the HTTP Cache Statistics section contains the following rows of information:

| Row Description | |
|---|---|
| Cacheable | This field analyzes the cacheable contents from the loaded trace. The percentage of Cacheable Responses is calculated from the amount of Cacheable content and the amount of total cache content. The percentage of Cacheable Bytes is calculated from the number of Cacheable bytes and total number of cache bytes. |
| Specified - No Store | This field analyzes the files from the loaded trace that are specified as “No Store”. The percentage of No Store Responses is calculated from the amount of No Store content and the amount of Total Cache content. The percentage of No Store Bytes is calculated from the number of No Store bytes and the total number of cache bytes. |

#### *5.2.5.14. Cache Simulation Results*

The Cache Simulation Results sub-category of the HTTP Cache Statistics section contains the Acceptable behavior, Duplicate File Download, and Duplicate File Download: Streaming sub-sections.

The following tables describe the rows of information in those sub-sections.

| Row | Description |
|---|---|
| Files downloaded once | The percentage of total responses and total bytes for files that were downloaded only once. This content is populated from the caching missed contents. |
| Files specified as "No-Store" | This content is calculated from the “No-Store” HTTP responses. The percentages are calculated from the cache diagnosis total and the number of total bytes. |

| | |
|---|---|
| Expired, but correct 304 responses sent from server | The percentage of total responses and total bytes for content with the HTTP response code 304. |
| Expired, downloaded again, but file has changed | The percentage of total responses and total bytes for content where the HTTP response has changed from the expired response. |

| Row | Description |
|---|---|
| Duplicate download (not expired) | The percentage of total responses and total bytes for content which is a duplicate download but has not expired. |
| Duplicate download (expired, but no "If-Modified Since" header sent) | The percentage of total responses and total bytes for content which is a duplicate download that has expired, and for which an “If-Modified-Since” header was not sent. |
| Duplicate download (expired, but "If-Modified Since" header ignored) | The percentage of total responses and total bytes for content which is a duplicate download that has expired and contains an “If-Modified-Since” header that was ignored. |

| Row | Description |
|---|---|
| Partial duplicate download (Not Expired) | The percentage of total responses and total bytes for content which is a partial duplicate download that has not expired. |
| Partial duplicate download (expired, but no "If-Modified Since" header sent) | The percentage of total responses and total bytes for content which is a partial duplicate download that has expired and for which an “If-Modified-Since” header was not sent. |
| Partial duplicate download (expired, but "If-Modified Since" header ignored) | The percentage of total responses and total bytes for content which is a partial duplicate download that has expired and for which an “If-Modified-Since” header was ignored. |
| | |

### 5.2.5.15. Duplicate File Analysis

The Duplicate File Analysis section of the Statistics Tab information page displays information about duplicate files that were downloaded during the trace.

The Duplicate File Analysis section contains the following information:

| Field | Description |
|---|---|
| Duplicate download (Cache not expired) | The percentage of total responses and total bytes for content which is a duplicate download, and for which the cache has not expired. These values are calculated with the total cache expiration count and cache expiration ratios. |
| **Field** | **Description** |
| Duplicate download (24 hr. cache not expired) | The percentage of total responses and total bytes for content which is a duplicate download, and for which the 24-hour cache has not expired. |
| Duplicate download (Cache expired) | The percentage of total responses and total bytes for content which is a duplicate download, and for which the cache has expired. |
| Duplicate download (24 hr. cache expired) | The percentage of total responses and total bytes for content which is a duplicate download, and for which the 24-hour cache has expired. |

### 5.2.5.16. Energy Efficiency Simulation

The Energy Efficiency Simulation section of the Statistics Tab information page displays the overall energy efficiency from the loaded trace. The section lists the amount of energy used for each of the different types of energy consumption that can affect the performance of the application, or the energy level of the device.

**Note:** The information displayed in this section depends on the type of device profile that is selected.

When a 3G device profile like *AT&T 3G* is selected, the Energy Efficiency Simulation section appears like the following:

**Energy Efficiency Simulation**

| | |
|---|---|
| DCH (Active): | 78.15 J |
| FACH (Standby): | 12.81 J |
| IDLE: | 0.00 J |
| IDLE->DCH (Active): | 3.14 J |
| FACH (Standby)->DCH (Active): | 2.68 J |
| DCH (Active) Tail: | 36.40 J |
| FACH (Standby) Tail: | 5.22 J |
| Total RRC Energy: | 96.78 J |
| Joules per Kilobyte: | 0.43 |
| GPS Active: | 0.00 J |
| GPS Standby: | 2.65 J |
| Total GPS Energy: | 2.65 J |
| Total Camera Energy: | 0.00 J |
| Bluetooth Active: | 0.00 J |
| Bluetooth Standby: | 0.00 J |
| Total Bluetooth Energy: | 0.00 J |
| Total Screen Energy: | 84.54 J |

When an LTE device profile like *AT&T LTE* is selected, the Energy Efficiency Simulation section appears like the following:

**Energy Efficiency Simulation**

| | |
|---|---|
| IDLE->Continuous Reception: | 0.94 J |
| Continuous Reception: | 19.43 J |
| Continuous Reception Tail: | 13.56 J |
| Short DRX: | 2.44 J |
| Long DRX: | 97.86 J |
| IDLE: | 0.54 J |
| Total RRC Energy: | 121.21 J |
| Joules per Kilobyte: | 0.53 |
| GPS Active: | 0.00 J |
| GPS Standby: | 2.65 J |
| Total GPS Energy: | 2.65 J |
| Total Camera Energy: | 0.00 J |
| Bluetooth Active: | 0.00 J |
| Bluetooth Standby: | 0.00 J |
| Total Bluetooth Energy: | 0.00 J |
| Total Screen Energy: | 76.98 J |

When a WiFi device profile like *AT&T WiFi* is selected, the Energy Efficiency Simulation section appears like the following:

**Energy Efficiency Simulation**

| | |
|---|---|
| WiFi Active: | 11.43 J |
| WiFi Tail: | 5.24 J |
| WiFi Idle: | 2.09 J |
| Total Wi-Fi Energy: | 13.52 J |
| GPS Active: | 0.00 J |
| GPS Standby: | 2.65 J |
| Total GPS Energy: | 2.65 J |
| Total Camera Energy: | 0.00 J |
| Bluetooth Active: | 0.00 J |
| Bluetooth Standby: | 0.00 J |
| Total Bluetooth Energy: | 0.00 J |
| Total Screen Energy: | 76.98 J |

The following table describes all the statistics contained in the Energy Efficiency Simulation section for either type of device profile:

| Field | Description |
|---|---|
| CELL_DCH (Active) | The total DCH time energy expended in the loaded traces. This is calculated from the RRC DCH time value and the power DCH value. |
| CELL_FACH (Standby) | The total FACH energy expended in the loaded trace. This is calculated from the RRC FACH time value and the power FACH value. |
| IDLE | The total idle time energy from the loaded trace. The idle energy should always be 0. |
| IDLE DCH (Active) | The amount of RRC IDLE to DCH (Active) state time energy consumption. |
| FACH (Standby) DCH (Active) | The amount of RRC FACH (Standby) to DCH (Active) time energy consumption. |
| DCH (Active) Tail | The amount of energy consumed during the RRC DCH (Active) Tail state period. |

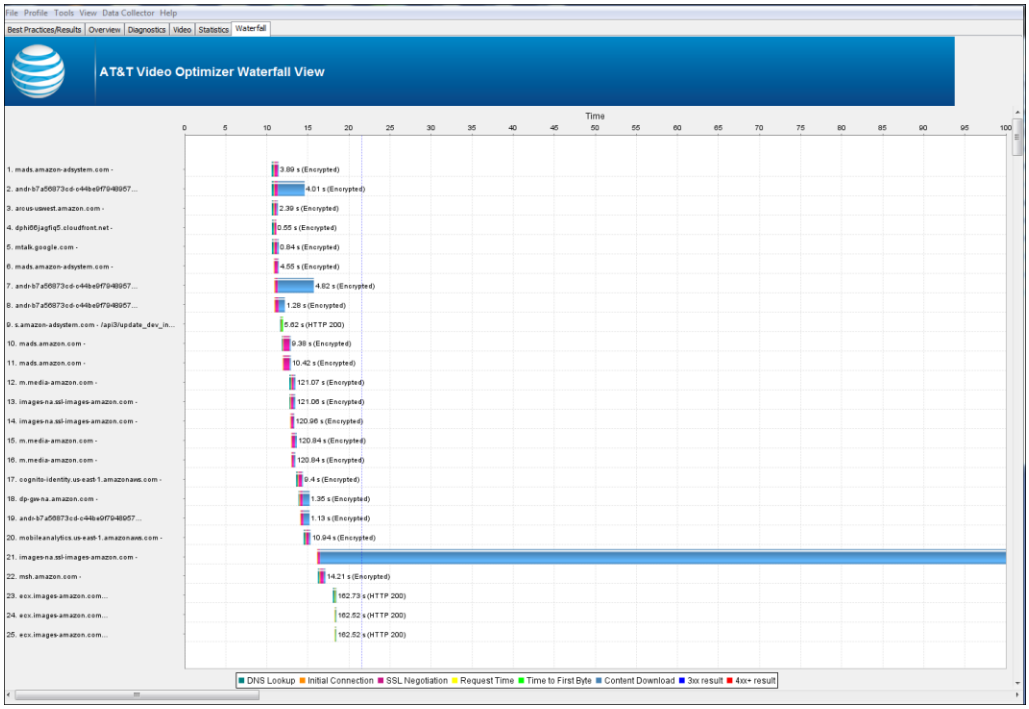
| Field | Description |
|---|---|
| FACH (Standby) Tail | The amount of energy consumed during the RRC FACH Tail state period. |
| IDLE Continuous Reception | The amount of energy consumed during all transitions from the IDLE state to Continuous Reception. |
| Continuous Reception | The amount of energy consumed during the Continuous Reception state. |
| Continuous Reception Tail | The amount of energy consumed during the Tail time of the Continuous Reception state. |
| Short DRX | The amount of energy consumed during the Short DRX state. |
| Long DRX | The amount of energy consumed during the Long DRX state. |
| WiFi Active | The amount of amount of energy consumed during the WiFi Active state. |
| WiFi Tail | The amount of amount of energy consumed during the WiFi Tail state. |
| WiFi Idle | The amount of amount of energy consumed during the WiFi Idle state. |
| Total RRC Energy | The sum of the CELL_DEH (Active), CELL_FACH (Standby), FACH (Standby) DCH (Active), IDLE DCH (Active), and IDLE energy consumption amounts. |
| Joules per Kilobyte | The number of Joules per Kilobyte from the loaded trace, calculated from the amount of total energy and total bytes. |

| | |
|---|---|
| GPS Active | The total energy consumed during the GPS Active state. In GPS Active state, the energy consumption will be equal to the time multiplied by the energy draw for Active GPS. |
| Long DRX | The amount of energy consumed during the Long DRX state. |
| WiFi Active | The amount of amount of energy consumed during the WiFi Active state. |
| WiFi Tail | The amount of amount of energy consumed during the WiFi Tail state. |
| WiFi Idle | The amount of amount of energy consumed during the WiFi Idle state. |
| Total RRC Energy | The sum of the CELL_DEH (Active), CELL_FACH (Standby), FACH (Standby) □ DCH (Active), IDLE □ DCH (Active), and IDLE energy consumption amounts. |

| Field | Description |
|---|---|
| Joules per Kilobyte | The number of Joules per Kilobyte from the loaded trace, calculated from the amount of total energy and total bytes. |
| GPS Active | The total energy consumed during the GPS Active state. In GPS Active state, the energy consumption will be equal to the time multiplied by the energy draw for Active GPS. |

### 5.2.6. Waterfall Tab

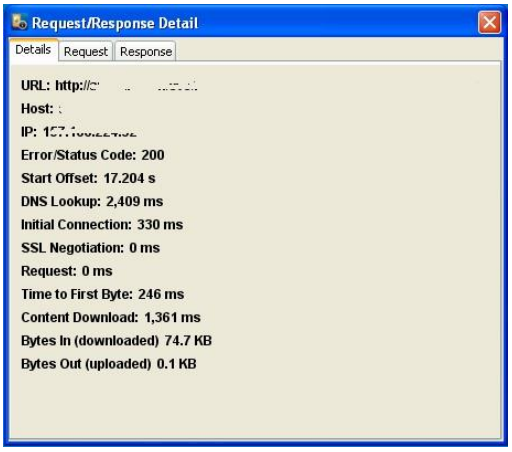
The Waterfall tab displays a waterfall view chart of the TCP connections from the trace spread over time. The chart can be expanded or contracted to get a detailed view into the connections in the trace data. The following image shows the Waterfall Tab. (Note: The URLs of the TCP connections have been deliberately hidden.)




1. As the key at the bottom of the tab indicates, the color-coded plots on the chart indicate the following information for each connection listed on the left side of the chart:
2.

| i. **Label** | ii. **Description** |
|---|---|
| iii. DNS Lookup | iv. The time until the DNS lookup was completed. |
| v. Initial Connection | vi. The time until the initial connection was made. |
| vii. SSL Negotiation | viii. The time spent in SSL negotiation. |
| ix. **Label** | x. **Description** |
| xi. Request Time | xii. The time spent requesting data from the server. |
| xiii. Time to First Byte | xiv. The time until the first byte was downloaded. |
| xv. Content Download | xvi. The time spent downloading the content. |
| xvii. 3xx result | xviii. An HTTP response code in the 300 range. |
| xix. 4xx+ result | xx. An HTTP response code in the 400 or 500 range. These codes indicate an error. |

3. When any of the color-coded plots on the chart are clicked, a Request/Response Detail dialog box (like the following) is shown:

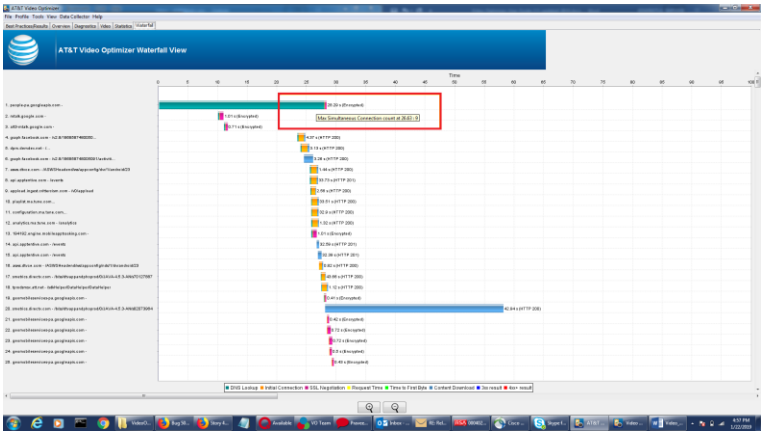



4. The tabs on the dialog box show details about the connection, information about the request, and the actual content of the response. The request/response data is like the detail provided in the Request/Response View table on the Diagnostics tab.

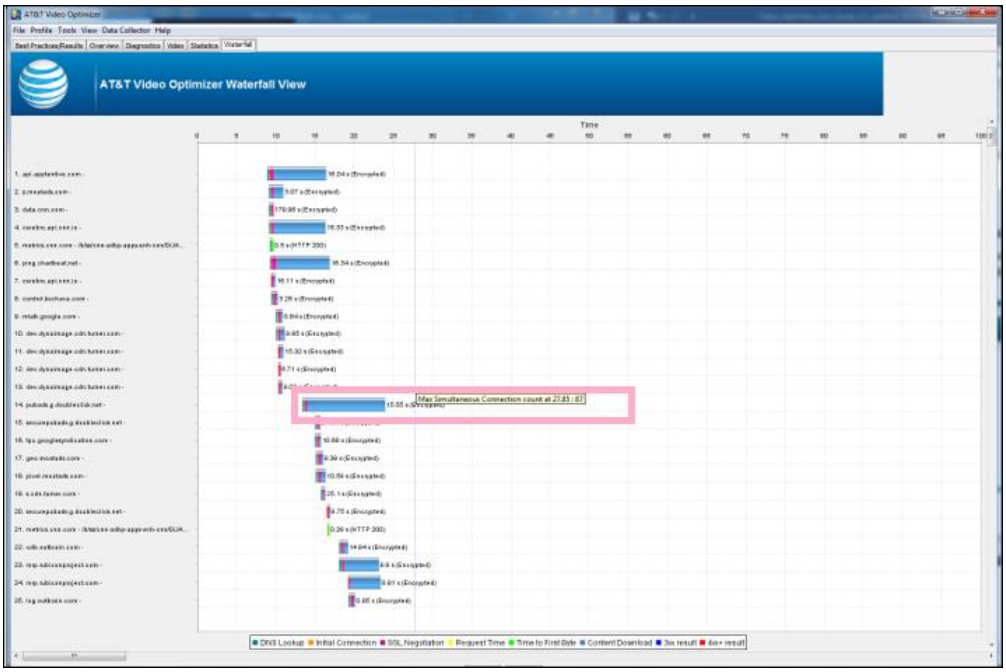
#### *5.2.6.1. Multiple Simultaneous connections to many end points*

Double clicking on the results section of the Multiple Simultaneous Connections to Many Endpoints best practice navigates to the waterfall tab.

The Waterfall tab displays appropriate domain names and IP addresses and shows the total connections at any point in time in the graph where a user hover. The Maximum Simultaneous Connections count for the trace is highlighted using a blue vertical line indicator at the moment when the most connections were open



Double clicking on the results section of multiple simultaneous connections navigates to the waterfall tab shows the maximum connections at that point.




# 6. Appendix I

The sections in Appendix I list the error messages for the Data Collector and for Video Optimizer.

## 6.1. Data Collector Error Messages

The following table lists and describes the error messages that can appear when using the Data Collector option in Video Optimizer.

| Error Message | Condition |
|---|---|
| Please enter trace folder name. | This error occurs if data collection is started without providing a trace folder name. |
| Trace folder name cannot have special characters or spaces. | This error occurs if data collection is started, and the trace folder name contains either non-alphanumeric characters, or spaces. |
| Trace folder already exists! Do you want to overwrite existing trace? | This error occurs if data collection is started, and the trace folder name already exists. When this error occurs, click Ok to proceed and overwrite the folder contents, or press Cancel to provide another name. |

| | |
|---|---|
| SD card is either not available or it is mounted. Please check the SD card before running Video OptimizerData Collector. | The Data Collector writes data files to a folder on the device's onboard SD card—but the SD card must not be mounted during data collection. This error occurs if no SD card is detected, or if the SD card is mounted when starting the Data Collector. |
| SD Card mounted, Video Optimizer Data Collector trace session terminated. | When the Data Collector finishes collecting data, it writes the data files to the device's onboard SD card—but the SD card must not be mounted during data collection. This error occurs if data collection is started, and the SD card is mounted. |
| SD Card memory full, stopping Video Optimizer Data Collector. | This error occurs during data collection, if the data collection process terminates because the SD card is full. |
| Video Optimizer Data Collector trace session can't be started with Flight Mode on and no active WIFI connection. | This error occurs if data collection is started when the device is in Flight Mode and there is no active WiFi connection. |
| **Error Message** | **Condition** |
| Video Optimizer Data Collector trace session stopped due to Flight Mode being turned on while there was no active WIFI connection. | This error occurs if Flight Mode is turned on while there is no active WiFi connection and data collection is taking place. |
| Video Optimizer Data Collector failed to start! | This message appears if an unexpected error occurs when the Start Collector button is clicked. |
| Video Optimizer Data Collector trace collection stopped. | This error occurs during data collection, if the data collection process terminates unexpectedly. |

| | |
|---|---|
| Video Optimizer Data Collector trace session can't be started with no active network connection. | This error occurs if a trace session is started with no active network connection. |

## 6.2. Video Optimizer Error Messages

The following table lists and describes the error messages that can appear when using the Video Optimizer.

| Error Message | Condition |
|---|---|
| iOS: HD: Video Optimizer is unable to match the project bundle ID from provisioning profile entered in the preference setting. Please update the project bundle ID from preference setting. | This error occurs when user has given incorrect bundle identifier in the preferences window. |
| iOS: HD: Video Optimizer is unable to match the provisioning profile from device selected. Please create a new provisioning profile from XCODE and update in preference setting. | This error occurs when user has used a different device to take a trace and not the one using which the provisioning profile got created. |
| iOS: HD: The provisioning Profile selected is invalid/expired. Please update the provisioning path in the preference setting. | This error occurs when the provisioning profile created in expired or the user connected a different iOS device for testing. |

| | |
|---|---|
| iOS: Remote Virtual Interface (rvi) error. | There was a disconnect between the mac interface and RVI. Please disconnect and reconnect the device or restart the device for continuation. |
| iOS: Error.rvi.reset connection | There was a disconnect between the mac interface and RVI. Please disconnect and reconnect the device or restart the device for continuation. |
| Missing Prerequisite Library: Unable to detect prerequisite dependency ideviceinstaller. Please verify. | This message occurs when one of the pre-requisites is not installed. |
| Missing Prerequisite Library: Unable to detect prerequisite dependency libimobiledeivice. Please verify. | This message occurs when one of the pre-requisites is not installed. |
| Missing Prerequisite Library: Unable to detect prerequisite dependency ifuse. Please verify. | This message occurs when one of the pre-requisites is not installed. |
| Video Optimizer requires WinPcap. Please install WinPcap which can be found at http://www.winpcap.org. | The Video Optimizer application is dependent on WinPcap, and it looks for the WinPcap installation during every launch. This error occurs if WinPcap is not found, or if Npcap is not installed in WinPcap API-compatible Mode |

| | |
|---|---|
| Video Optimizer requires AppleQuartz renderer. Please set apple.awt.graphics .UseQuartz flag to true | This error occurs if the Video Optimizer is being used on a Mac OS, and the apple.awt.graphics.UseQuartz flag is not set to true. |
| Unexpected Exception: exception message | This error occurs if Video Optimizer encounters an unexpected exception. |
| Trace too big to load is displayed to the user. | This error occurs if Video Optimizer reaches the maximum amount of memory allocated for it. |
| Video Optimizer is unable to open a file of this size.<br>Please Save As... to a local disk and open from there. | This error occurs if the opening of a trace file will force Video Optimizer to reach the maximum amount of memory allocated for it. |

| Error Message | Condition |
|---|---|
| Video Optimizer is unable to open a file extension type file. Please Save As... to a local disk and open from there. | This error occurs if the type of the file that is being opened is not recognized by Video Optimizer. |
| Invalid trace in directory: trace folder path and exception message. | This error occurs if Video Optimizer encounters data in a trace file that is invalid when performing analysis. |

| | |
|---|---|
| An error occurred when trying to save the chart. | Occurs if there is an error while saving the Diagnosis Chart to an image file. |
| No trace loaded. Please load trace files before selecting this option. | This error occurs if one of the following menu options is selected before a trace file has been opened in the Video Optimizer:<br>• Time Range Analysis (Tools Menu)<br>• PCAP File Analysis (Tools Menu)<br>• Select Applications/IPs (View Menu) |
| Video Optimizer could not find trace folders in selected path. Please select a valid trace folder path. | This error occurs when a folder that does not contain valid trace files is selected when using the Open Trace or Data Dump menu options. |
| The USB device got disconnected. Please check the connection. | Occurs when a device that is connected via USB to a device running the Data Collector, is disconnected unexpectedly. |
| Video Optimizer Collector is already running on the device. Please stop it and try again. | Occurs when an instance of the Data Collector is running on a device while another instance is started. |
| Time values must be numeric. | Occurs in the Time Range Analysis Dialog if a nonnumeric value is entered for the Start or End time. |
| Start time must be less than End Time. | Occurs in the Time Range Analysis Dialog if a Start Time value is entered that is greater than the End Time value. |

| Error Message | Condition |
|---|---|
| Start Time and End Time must be between 0.00 and trace length. | Occurs in the Time Range Analysis Dialog if time value is entered that is less than 0 or greater than the total time of the trace. |
| The system is unable to open .csv files by default. Please set a default program for .csv files. | Occurs if the Export option is selected to save one of the charts or tables in the Overview, Diagnosis and Statistics tabs to the .csv format, and there is no default program set in the system for .csv files. |

| | |
|---|---|
| Unable to connect to printer: printer name. | Occurs when Video Optimizer is unable to connect to a printer when the Print option is selected in the File menu. |
| Error in parsing alarm analysis info. | Occurs when Video Optimizer is unable to parse information about an alarm on the device. |
| More than one device or emulator is connected to PC. | Occurs if the computer is connected to multiple devices of device emulators, when the Start Collector option on the Data Collector menu is selected. |
| Could not find device or emulator connection. Please verify the connection and ADB daemon is started. | Occurs if the user is not connected to a device or device emulator, or if the ADB daemon is not started, when the Start Collector option on the Data Collector menu is selected. |
| An unexpected error has occurred, please restart the device. | Occurs when there is an unexpected error in the device emulator. |
| Connection to device or emulator is lost. Please wait for some time before starting data collector. | Occurs if there is an error with the ADB connection. |
| Trace directory already exists. Do you want to overwrite trace files in the directory? | Occurs if an existing Trace directory name is entered in the dialog box when the Start Collector menu option is selected on the Data Collector menu. |
| Unable to create the Emulator trace directory. | Occurs when the trace directory cannot be created. |

| Error Message | Condition |
|---|---|
| Trace name which you want to replace is currently loaded. Do you want to clear the trace? | Occurs if the name of the currently loaded Trace is entered in the dialog box when the Start Collector menu option is selected on the Data Collector menu. |

| | |
|---|---|
| Error starting Video Optimizer Data Collector. | Occurs if there is an error when the Start Collector menu option is selected on the Data Collector menu. |
| Error tcpdump not compiled for this device. | Occurs if the device is not supported by the Data Collector. |
| Error stopping Video Optimizer Data Collector. | Occurs if there is an error when stopping the Video Optimizer Data Collector (using the Stop Collector menu option on the Data Collector menu) after it has been started from Video Optimizer. |
| Error pulling Video Optimizer Data Collector traces. | Occurs if there is an error when pulling Trace files from the Video Optimizer Data Collector to the local system. |
| Video Optimizer Analyzer stopped unexpectedly. | Occurs if the Data Analyzer unexpectedly stops. |
| ADB Rejected the Video Optimizer Data Collector device Connection. | Occurs if the Android Debug Bridge (ADB) cannot connect to the device. |
| Emulator SD card is full. Please free some space to start Video Optimizer Data Collector. | Occurs if the Emulator does not have enough space on its SD card to save the trace files collected by the Video Optimizer Data Collector. |
| Please set your device USB Mode to "Charge Only" - otherwise the SD Card is not available but is required by application. | Occurs if the Emulator does not have an SD card available. An SD card is required by the Video Optimizer Data Collector when using the Emulator. |
| Video Optimizer requires a virtual SD card to be configured when using the Android Emulator. | Occurs if no virtual SD card was configured before using the Android Emulator. |

| Error Message | Condition |
|---|---|

| | |
|---|---|
| Emulator SD card does not have enough space; it must have 5 MB or more. | Occurs if the Emulator does not have at least 5MB of space available on its SD card to save the trace files collected by the Video Optimizer Data Collector. |
| Device SD card does not have enough space; it must have 5 MB or more. | Occurs if a device does not have at least 5MB of space available on its SD card to save the trace files collected by the Video Optimizer Data Collector. |
| Emulator SD Card memory full, stopping Video Optimizer Data Collector. | Occurs if the Emulator does not have enough any memory remaining on its SD card. When this error occurs the Video Optimizer Data Collector is stopped. |
| Device SD Card memory full, stopping Video Optimizer Data Collector. | Occurs if the Device does not have enough any memory remaining on its SD card. When this error occurs the Video Optimizer Data Collector is stopped. |
| Unexpected error accessing emulator SD Card: exception message | Occurs if there is an unexpected error while the Video Optimizer Data Collector is accessing the Emulator. |
| Unexpected error accessing device SD Card: exception message | Occurs if there is an unexpected error while the Video Optimizer Data Collector is accessing the SD Card of a device. |
| Device SD card is not available but is required by application. | Occurs when the Video Optimizer Data Collector attempts to access the SD Card of a device when it is not available. |
| Trace folder name should not contain special characters or spaces. | When the Start Collector menu option is selected on the Data Collector menu, the user is prompted to enter a Trace folder name. This error occurs if the Trace folder name contains an invalid special character or a space. The folder name can only contain alphanumeric characters or a (-) special character. |
| Trace folder name should not be more than 50 characters. | Occurs if the Trace folder name that is entered in the dialog box when the Start Collector menu option is selected on the Data Collector menu, is longer than 50 characters. |

| Error Message | Condition |
|---|---|

| | |
|---|---|
| Emulator error with tcpdump/key.db push. | Occurs if there is an error while transferring the collected trace files from the device or device emulator to the local system.<br><br>This transfer is initiated by the Video Optimizer Data Analyzer when the Stop Collector menu option is selected on the Data Collector Menu, and the Pull Traces menu option is selected on the Data Collector. |
| Emulator I/O exception caused data collector failure. | Occurs if there is an Input / Output exception when the Data Collector tries to connect to the device or device emulator.<br><br>The Data Collector is started from the Data Analyzer by selecting the Start Collector option on the Data Collector menu |
| No application found to open PCAP trace. Please install an application like WireShark for PCAP analysis. | Occurs if the PCAP File Analysis option is selected on the Tools menu, but an external tool (like WireShark) for analyzing PCAP files is not installed. |
| No traffic.cap file found in trace. | Occurs if the PCAP File Analysis option is selected on the Tools menu, but a traffic.cap file is not found in the loaded trace folder. |
| Not able to start. | Occurs if the APK is unable to start. |
| Video file is not valid. | Occurs if the Video Optimizer Image/Video Viewer attempts to load an invalid video file, or if a trace is loaded that contains an invalid video file. |
| Unable to read file. | Occurs if the Video Optimizer Image/Video Viewer attempts to load a video file that it is unable to read, or if a trace is loaded that contains a video file that Video Optimizer is unable to read. |
| Video display conversion of video.mp4 to video.mov file failed. | Occurs if the Video Optimizer Data Analyzer fails while converting the video file from .MP4 to .MOV. |
| ERROR: Trace directory is empty | Occurs if the Video Optimizer Image/Video Viewer attempts to load a video file, but the Trace directory is empty. |
| ERROR: Input file does not exist; nothing to convert. | Occurs if the .MP4 video file does not exist when the Video Optimizer Image/Video Viewer is attempting to convert it to .MOV. |

| Error Message | Condition |
|---|---|

| | |
|---|---|
| ERROR: No permission to write to output file for conversion. | Occurs if the Video Optimizer Image/Video Viewer does not have permission to write the output file, when it is converting the .MP4 video to the .MOV format. |
| ERROR: Input file is a directory; cannot be converted. | Occurs if the Video Optimizer Image/Video Viewer cannot convert the input file (an .MP4 video) to the .MOV format. |
| ERROR: Output file is a directory; cannot be converted. | Occurs if the Video Optimizer Image/Video Viewer encounters a directory name instead of a file name when it is converting the .MP4 video to.MOV format. |
| ERROR: Output file still exists after deletion; cannot be converted. | Occurs if the Video Optimizer Image/Video Viewer encounters an error when preparing the output file for conversion from .MP4 to.MOV. |
| ERROR: Unable to read file. | Occurs if a file that was placed in the trace directory cannot be read by the Video Optimizer. |
| ERROR: Video display conversion of video.mp4 to video.mov file failed. | Occurs if the Video Optimizer Image/Video Viewer encounters an error when converting the output file from .MP4 to.MOV. |
| ERROR: Exception setting up video player. | Occurs if there is an exception when the Video Optimizer Image/Video Viewer is initializing. |
| Video is already Synched. Do you want to Re-Sync again? | Occurs if the Sync Video button is clicked when the video is already in sync. |
| The previous sync point has been cleared. In order to Re-sync the video, select the desired time point in the Video player and then press the Sync Video button now. | Occurs if the Sync Video button is clicked when no sync points have been set. |

| Error Message | Condition |
|---|---|

| | |
|---|---|
| ERROR: Multiple external video files exist in the trace folder. Please add only one external video and remove the rest. | Occurs when there is more than one video file in the trace folder when the trace is loaded. |
| Category String Error | Occurs when an invalid string is entered for the name of a User Defined Burst. |
| Unexpected error exporting table | Occurs when there is an error exporting the Burst Analysis Table. |
| Error loading the list of profiles. | Occurs when there is an error loading a device profile. |
| Error setting the selected profile to the Video Optimizer. | Occurs when there is an error setting a device profile in selected in the Select Device Profile dialog box that is opened by the Load option in the Profile menu. |
| Error loading last device profile. Default device profile is being used. | Occurs if there is an error loading the device profile. In this case, the default profile will be used instead. |
| Error reading device profile attributes: attribute name. | Occurs when a Device Profile is loaded that contains an unrecognized attribute name. |
| Unable to save file due to the errors below: file error(s). | Occurs when there are file errors while attempting to save a customized Device Profile. |
| Unable to open file due to errors below: file error(s). | Occurs when there are file errors while attempting to open a Device Profile. |
| Unable to load file due to errors below: file error(s). | Occurs when there are file errors while attempting to load a Device Profile. |
| Error writing to file: file error(s). | Occurs when there is an error writing to the file selected in the file chooser dialog. |
| Unable to load content. Download may have been interrupted. | Occurs when there is an error opening content in the Video Optimizer Image/Video Viewer. |
| Microsoft Network Monitor related error. | Occurs when there is an error related to the Microsoft Network Monitor. |
| **Error Message** | **Condition** |

| | |
|---|---|
| Could not load Microsoft Network Monitor trace file. | Occurs when there is an error loading a trace file that was collected by the Microsoft Network Monitor. |
| Video Optimizer was unable to open the file. It may be necessary to install Microsoft Network Monitor. | Occurs when a trace file that was collected by the Microsoft Network Monitor is opened, but Microsoft Network Monitor is not installed. |
| Timeout in starting the collector trace. | Occurs when the trace file is so large that it time out before it can be loaded by the Analyzer. |
| Video Optimizer Collector is not installed on the device. | Occurs when the Start Collector menu option is selected but the Data Collector apk is not installed on the device. |
| Video Optimizer collector is not started. Its current activity has been brought to the front. Please exit the activity and try again. | Occurs when activity by the Collector is displayed in front of the main start screen before the Collector is started. |
| Video Optimizer collector is not stopped. Please hide the activity on the device and press OK. | Occurs when the Collector is not stopped, and activity on the device is displayed in front of the main screen. |
| Video Optimizer could not find a trace in the selected folder. Please select a valid trace folder. | Occurs when a trace folder is selected (using the Open Trace menu option) that does not contain valid trace files. |

## 6.3. Glossary

This following table contains a list of Mobile Web-associated terms, with their associated definitions. For a more comprehensive list, see the World Wide Web Consortium (W3C).

| Term | Definition |
|---|---|
| Age | A property of a Response Entity. The length of the elapsed time since the Entity was either Served by the Origin Server, or successfully validated. |

| | |
|---|---|
| Average Rate | The amount of data in KB over the time the trace was run. Apps that stream content should score high here, apps with few connections should score lower. |
| Burst | Consecutive packets of data transferred in a batch over a TCP connection. Bursts can be initiated by the user, the app, or the network. |
| Cache | A local process implemented in the client that creates copies of Response Messages and serves them to the client on the Server's behalf, if it remains identical to the Origin Server's copy. When used properly, the use of Response Caches significantly reduces application response time and bandwidth consumption. |
| Cacheable | A response is cacheable if the requirements of the Request Method, Request Header Fields, and the Response Status indicate that it is cacheable. |
| Client | A program that establishes connections for sending requests. |
| Connection | A virtual circuit, established at the Transport Layer, that is used to connect two programs so that they can communicate using TCP. |
| Content Negotiation | The mechanism for selecting the appropriate representation for servicing a request. The representation of entities in any response can be negotiated (including Error Responses). |
| Core Network | The Internet backbone. The network that the Radio Access Network is connected to. |
| Energy Consumption | As your application becomes more efficient, the J/KB should decrease. This means you are consuming less battery energy per kilobyte. |
| Entity | The requested content. Delivered as the payload of Response/Request messages. Request and |

| Term | Definition |
|---|---|
| | Response messages do not always carry a payload. An Entity consists of entity-header that contains meta-information and an entity-body that contains web content, although some responses will include only the entity-headers. |

| | |
|---|---|
| Explicit Expiration Time | The expiration time associated with an Entity—when specified by the Origin Server.<br>Beyond that point in time, the Cache can continue serving the local copy of the Entity, but only if it passes a Validation test. |
| File Types | A breakdown of all files seen during the trace (in bytes). Files sent through HTTPs are listed as Encrypted. |
| First-Hand | A property of a Response. A response that is received directly from the Origin Server. Cached Responses are copies of First-Hand Responses. |
| Fresh | A property of a Response Entity. Indicates that a Cached Response is still implicitly valid. A Fresh Response is a response that has not exceeded its Freshness Lifetime. |
| Freshness Lifetime | A property of a Response Entity. The period in which a cacheable response remains implicitly valid.<br>How long it takes for a cacheable response to reach its expiration time. |
| Gateway | A Server that acts as an intermediary for another server.<br>Unlike proxies, gateways behave like Origin Servers, receiving resource requests. The requesting client cannot be aware that it is communicating with a gateway. |
| GPRS | General Packet Radio Services.<br>Single GSM error-corrected circuit-switched data channel. |
| Heuristic Expiration Time | The expiration time associated with an entity—determined programmatically by Cache management logic.<br>A Cache management strategy used whenever the Origin Server doesn't specify an Explicit Expiration Time. |
| HSDPA | High-speed Downlink Packet Access. |

| Term | Definition |
|---|---|

| | |
|---|---|
| | In the same way that EDGE uses techniques to increase speeds over GPRS, HSDPA employs these same techniques, as well as others, to increase the speed of UMTS data channels. Also known as UMTS/HSDPA. UMTS/HSDPA is a wide-area wireless data service |
| HTTP | Application level, stateless, communication protocol.<br>Client/server communication consists solely of independent pairs of Requests and Responses. |
| Inbound/Outbound | Terms indicating the path of Request and Response messages.<br>Inbound messages travel toward the Origin Server.<br>Outbound messages travel toward the User Agent. |
| Joules | The SI unit of energy.<br>The work required to produce one watt of power for one second (think of Kilowatt Hours). |
| Long Burst Connection | A long burst is one that sends a large amount of data in a short period of time. If most of the data is consumed, this is a good way to send data (see tightly grouped above). |
| Message | The basic unit of HTTP communication. |
| MIME | Multipurpose Internet Mail Extensions. |
| Non-Periodic Connections | Connections that recur periodically can cause rapid battery drainage. Consider if your periodic pings are required, if the timing could be lengthened, or if other alternatives exist (Pushing alerts is more efficient than regular polling). |
| Origin Server | The server on which a given Resource either resides or is created. |
| Pcap | Packet Capture<br>An API for capturing network traffic.<br>Unix-like systems implement Pcap in the libpcap library; Windows uses a port of libpcap known as WinPcap. |
| Proper Session Termination | The percentage of connections that close immediately with no delay. Connections that close in a delayed fashion, keep the RRC state Machine on longer - needlessly draining the battery. |
| Proxy | A program that acts as both a server and a client for making requests on behalf of other clients. |

| Term | Definition |
|---|---|
| | Requests are either serviced internally, or are passed on with possible translation, to other servers.<br>A proxy MUST implement both the client and server HTTP requirements.<br>A transparent proxy is a proxy that does not modify the request or response beyond what is required for authentication and identification.<br>A non-transparent proxy is a proxy that modifies the request or response to provide some added service to the user agent, such as group annotation services, media type transformation, protocol reduction, or anonymity filtering.<br>Except where either transparent or non-transparent behavior is explicitly stated, the HTTP proxy requirements apply to both types of proxies. |
| Radio Access Network (RAN) | The UMTS wireless network, connecting mobile devices to the Core Network. |
| Representation | A Response Entity that is subject to Content Negotiation.<br>Multiple representations can be associated with a Response Status. |
| Request | A request message from a client to a server includes, within the first line of that message, the method to be applied to the resource, the identifier of the resource, and the protocol version in use. |
| Resource | Any network Data Object or Service that can be identified by a URI.<br>Resources can be made available in multiple representations (e.g. multiple languages, data formats, size, and resolutions) or vary in other ways. |
| Response | After receiving and interpreting a Request Message, a web server fulfills the request by sending back an HTTP Response Message, which contains the requested content as the payload. |

| | |
|---|---|
| Semantically Transparent | A property that describes the way a Cache behaves. In terms of content quality, content served from a Cache that is Semantically Transparent matches that served from the Origin Server.<br>Except for the addition of hop-by-hop headers, the client receives Responses that are identical to First-Hand Responses. |

| Term | Definition |
|---|---|
| Server | A program that accepts connections to service requests and sends back responses.<br>Any given program can be capable of being both a client and a server; our use of these terms refers only to the role being performed by the program for a connection, rather than to the program's capabilities in general. Likewise, any server can act as an origin server, proxy, gateway, or tunnel, switching behavior based on the nature of each request. |
| Session | HTTP Session. The conversation that takes place, between a client and a server.<br>Initiated by the client but concluded by either the client or the server.<br>Consists of a series of network Request-Response transactions.<br>Lasts for the duration of the conversation, usually minutes. |
| Session Termination | This graph is scoring the types and actions of the TCP connections found in this trace. |
| Signaling overhead | The higher the percentile, better the performance of your application. Signaling overhead indicates the time spent in RRC state transitions. The lower the signaling overhead number, lower the count of state transitions, and the higher your percentile rank. |
| Simulation | Based on device profile. |
| Stale | A property of a response entity that indicates that a cached response is no longer implicitly valid. A stale response is a response that has exceeded its freshness lifetime. |
| Stateful | In a session, at least one of the communicating parties needs to save information about the session history to be able to communicate. |

| | |
|---|---|
| Stateless | The communication consists of independent requests and responses. |
| Tightly Grouped Connections | Connections that are grouped together efficiently use the radio while it is turned on. Connections that are spread out keep the radio on for a longer period, adding to the battery drain. |
| Trace | A record of the Information generated by a device and by the network communicating with the device. Trace information can include packets transferred |

| Term | Definition |
|---|---|
| | between the device and network, radio energy usage information, user input information, device information, and information from peripheral applications. |
| Trace Benchmark | This graph benchmarks your trace to the results of traces run on top mobile applications. This gives you an idea of where your application stands in comparison to other applications. The rankings here do not signify anything specific other than a ranking. |
| Tunnel | An intermediary program that acts as a blind relay in the Client/Server connection.<br>Once active, a Tunnel is not considered a party to the HTTP communication.<br>Tunnels are initiated by an HTTP Request. A tunnel ceases to exist when the connections terminates. |
| UMTS | Universal Mobile Telecommunications System. The 3G version of the GPRS technology. Based on GSM. UMTS radio link. |
| UMTS Data Channel | The link established between the mobile device and the cell tower. |
| Upstream/Downstream | Terms that describe the direction in which messages flow.<br>Messages travel from Upstream, to Downstream. |
| User Agent | The client that initiates a Request.<br>Clients are usually End-user programs, such as Web Browsers, but they can also be Service programs such as Spiders (web-traversing robots). |
| User Agent | A client application. Usually implements HTTP 1.1, to communicate with a web server. |

| | |
|---|---|
| Validator | Timestamp information that accompanies Response Entities.<br>Stored a Response Entity header field.<br>Used to validate Stale cache entries.<br>When an Origin Server sends a Full Response, it includes a Validator in the Entity-header, which along with the Entity-body, becomes a local cache entry.<br>A Client (user agent or proxy cache) makes a Conditional Request for a cached copy a Resource when it must guarantee the Resource's validity. The Server evaluates the condition based on the result of comparing the value of its local copy of the |
| **Term** | **Definition** |
| | requested resource's validator (the current version of the Resource), against that of the value of one in the Request. If they match, it responds with a special status code (usually, 304 (Not Modified)) and no entity-body. Otherwise, it returns a Full Response.<br>Thus, we avoid transmitting the full response if the validator matches, and we avoid an extra round trip if it does not match.<br>A protocol element (e.g., an entity tag or a LastModified time) that is used to determine whether a Cache Entry is usable (i.e., an equivalent copy of an entity. |
| Variant | At a given instant, Resources can have multiple Representation(s).<br>Each Representation is referred to as Variant. Use of the term Variant does not necessarily imply that the resource is subject to Content Negotiation. |

# 7. Appendix II

The sections in Appendix II describe how to use the legacy Rooted Data Collector APK.

## 7.1. Rooted Data Collector APK

Rooted collection is a legacy product and is no longer updated. It is still included in the Video Optimizer package, but if you encounter issues, they are unlikely to be resolved in a future release.

### 7.1.1. Prerequisites for using the Rooted Data Collector APK

To collect an application trace using the Data Collector APK, you need the following:

- A Video Optimizer Data Collector APK (included in the install on your computer)
- An Android test device running Android version between 4.x and 6.0 (rooted Data Collector does not work on Marshmallow and above)

### 7.1.2. Installing Rooted Data Collector APK

1. Start a trace as described in Using Video Optimizer to Collect Data, selecting the rooted option.
2. The apk will be installed on your phone.

#### *7.1.2.1. Collecting a Trace Using the Data Collector*

Once you have installed the Video Optimizer on your test device, you can operate it directly on the device to collect a trace; or for devices that do not allow you to capture trace video, you can operate the Data Collector via a USB connection using the commands on the Start Collector menu in Video Optimizer.

**To operate the Data Collector from Video Optimizer, do the following:**

Step 1. Select Preferences from the File menu.
Step 2. Select the General tab.
Step 3. For the Adb Path field, browse to the directory where the Android Debug Bridge (ADB) executable is located, and select it.
Step 4. Click Save and Close.
Step 5. Ensure that you have a USB connection between the test device and the computer where Video Optimizer is installed.
Step 6. Select Start Collector from the Data Collector menu.
Step 7. On the test device, select OK when the “Allow USB Debugging” prompt asks you to confirm the RSA key on the test device.
Step 8. Run testing scenarios on your app while Video Optimizer is collecting data. For example, launch the app, exercise the main functions of the app, and close the app.
Step 9. Select Stop Collector from the Data Collector menu

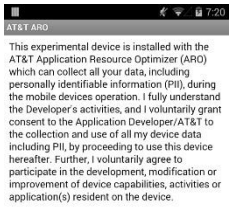
The trace data, including video, will be transferred to the computer via USB as part of the trace collection process.

**Note:** This method allows you to capture a video of the trace via USB for Android devices that do not support the Video Optimizer Data Collector capturing trace video directly.

**To operate the Video Optimizer Data Collector directly on an Android test device, do the following:**

Step 1. Open the program list on your device (**Figure**), find the “Video Optimizer” icon, and launch it.


Step 2. Click Accept on the Legal Terms screen (**Figure**) to proceed to the main screen of Video Optimizer.





Step 3. Use the controls on the Video Optimizer main screen to cancel any running applications that you do not want to test.

Step 4. Select whether to record video with your trace and start the Data Collector.

Step 5. Click Open Task Killer to stop running tasks and select the tasks that you want to stop before you start collecting data. This ensures that you are collecting trace data only from the application that you want to test.

Step 6. Click Record Video to record video while you are capturing the trace.

Step 7. Click Start Collector to start the Data Collector.