

Introduction

This document describes how to setup WireShark as a temporary MCU Syslog server for troubleshooting. Even though this guide is written towards use with an MCU, the same principals apply to all devices using WireShark as a temporary Syslog server.

Contributed by Matt Limbrick, Cisco TAC Engineer.

Requirements

Cisco recommends that you have knowledge of these topics:

- Wireshark "display filters" and "capture filters"
- Codian MCU menu navigation

Components Used

The information in this document is based on these software and hardware versions:

- Windows 7 VM hosting Wireshark (ver 1.12.7) IP: 14.80.98.182
- Codian 8510 MCU ver 4.5(1.55) IP: 14.80.76.9

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

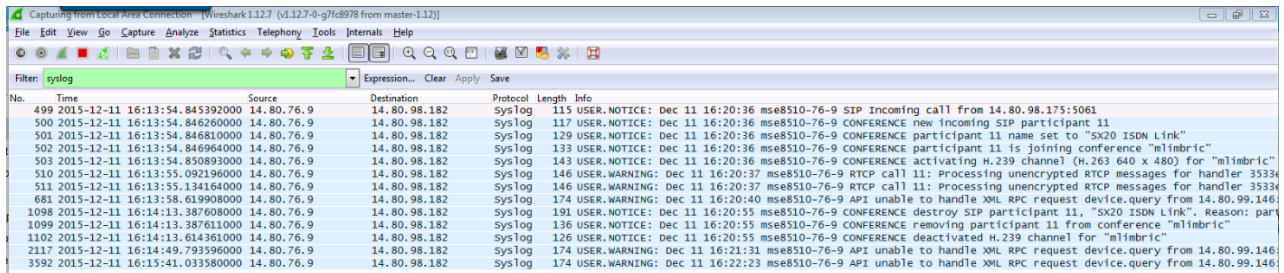
Wireshark as a Syslog server

Wireshark is a network protocol analyzer that allows you to run packet captures on a computer running the application. As Wireshark can be used to capture all types of traffic, it can be handy to use for a temporary Syslog server when you are without one. Most of the time, customers will already have this tool installed on their PC which prevents having to find and download a syslog server and get approval. To get started, you will 1st need to download and install Wireshark

<https://www.wireshark.org/download.html>

The simplest way to use Wireshark is to point the Syslog configuration of your device to a PC running Wireshark. Use the "display filter" of "syslog" to see the results and then export this data as a packet capture or as a text file.

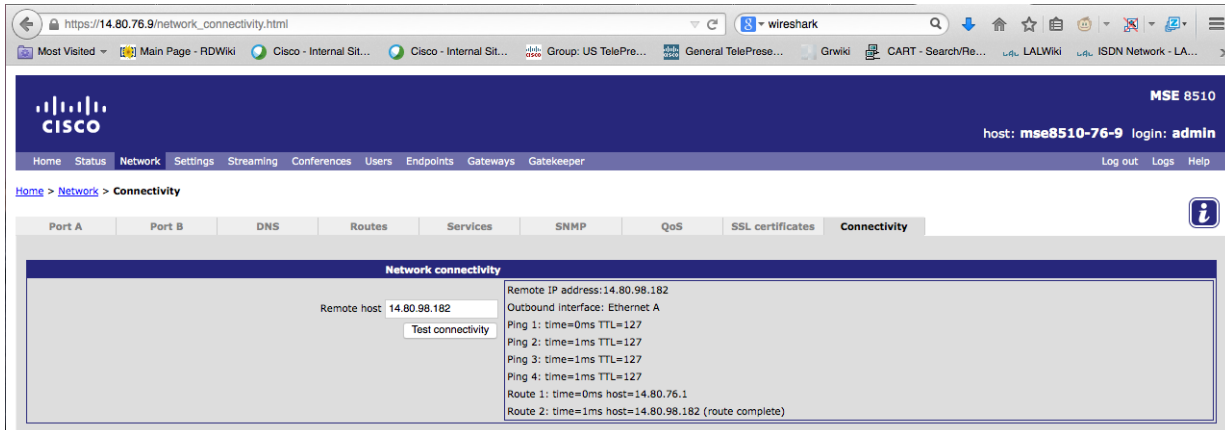
As you can see below, I have captured a SIP call connecting to a conference and then disconnecting. At this point, I do not have the log levels turned up so there is minimal details here.



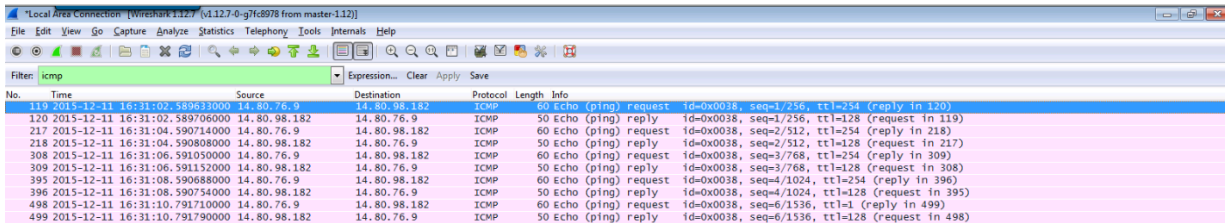
No.	Time	Source	Destination	Protocol	Length	Info
499	2015-12-11 16:13:54.845392000	14.80.76.9	14.80.98.182	Syslog	115	USER.NOTICE: Dec 11 16:20:36 mse8510-76-9 SIP incoming call from 14.80.98.175:5061
500	2015-12-11 16:13:54.846260000	14.80.76.9	14.80.98.182	Syslog	117	USER.NOTICE: Dec 11 16:20:36 mse8510-76-9 CONFERENCE new incoming SIP participant 11
501	2015-12-11 16:13:54.846810000	14.80.76.9	14.80.98.182	Syslog	129	USER.NOTICE: Dec 11 16:20:36 mse8510-76-9 CONFERENCE participant 11 name set to "SX20 ISDN Link"
502	2015-12-11 16:13:54.846964000	14.80.76.9	14.80.98.182	Syslog	133	USER.NOTICE: Dec 11 16:20:36 mse8510-76-9 CONFERENCE participant 11 is joining conference "mlmbric"
503	2015-12-11 16:13:54.850893000	14.80.76.9	14.80.98.182	Syslog	143	USER.NOTICE: Dec 11 16:20:36 mse8510-76-9 CONFERENCE activating H.239 channel (H.263 640 x 480) for "mlmbric"
510	2015-12-11 16:13:55.092196000	14.80.76.9	14.80.98.182	Syslog	146	USER.WARNING: Dec 11 16:20:37 mse8510-76-9 RTP call 11: Processing unencrypted RTP messages for handler 35334
511	2015-12-11 16:13:55.134164000	14.80.76.9	14.80.98.182	Syslog	146	USER.WARNING: Dec 11 16:20:37 mse8510-76-9 RTP call 11: Processing unencrypted RTP messages for handler 35334
681	2015-12-11 16:13:58.619908000	14.80.76.9	14.80.98.182	Syslog	174	USER.WARNING: Dec 11 16:20:40 mse8510-76-9 API unable to handle XML RPC request device.query from 14.80.99.146:
1098	2015-12-11 16:14:13.387608000	14.80.76.9	14.80.98.182	Syslog	191	USER.NOTICE: Dec 11 16:20:55 mse8510-76-9 CONFERENCE destroy SIP participant 11, "SX20 ISDN Link". Reason: part
1099	2015-12-11 16:14:13.387611000	14.80.76.9	14.80.98.182	Syslog	136	USER.NOTICE: Dec 11 16:20:55 mse8510-76-9 CONFERENCE removing participant 11 from conference "mlmbric"
1102	2015-12-11 16:14:13.614361000	14.80.76.9	14.80.98.182	Syslog	126	USER.NOTICE: Dec 11 16:20:55 mse8510-76-9 CONFERENCE deactivated H.239 channel for "mlmbric"
2117	2015-12-11 16:14:49.793596000	14.80.76.9	14.80.98.182	Syslog	174	USER.WARNING: Dec 11 16:21:31 mse8510-76-9 API unable to handle XML RPC request device.query from 14.80.99.146:
3592	2015-12-11 16:15:41.033580000	14.80.76.9	14.80.98.182	Syslog	174	USER.WARNING: Dec 11 16:22:23 mse8510-76-9 API unable to handle XML RPC request device.query from 14.80.99.146:

Using WireShark with an MCU as a Syslog server

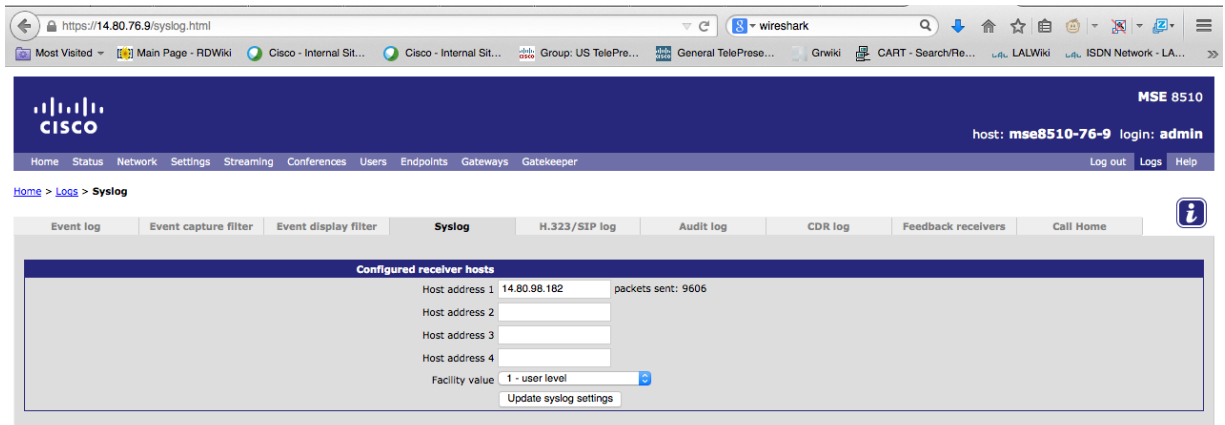
To use WireShark as a Syslog server, you want to 1st be sure that the computer hosting WireShark is able to receive packets from the MCU. To do so, start WireShark with "display filter" of "icmp". From the (MCU > Network > Connectivity tab), type the IP of the computer hosting WireShark and click on "Test Connectivity". (i.e. 14.80.98.182)



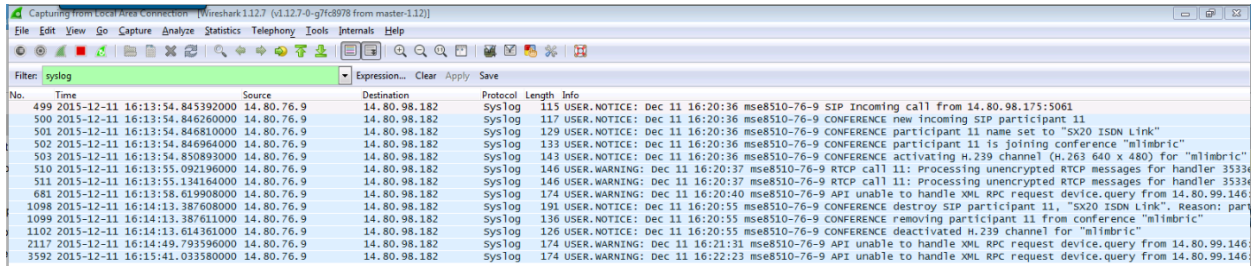
In the WireShark application you should now see the ICMP (ping) packets from the MCU (i.e. 14.80.76.9) while using the "display filter" of "icmp".



Now that you have verified packets from the MCU can route to the computer hosting WireShark, we can configure the syslog settings of the MCU (or other device) to point to WireShark. On the MCU, go to (MCU > Logs > Syslog tab) and apply the IP address of the computer hosting WireShark (i.e. 14.80.98.182), then click "Update syslog settings". The option for Facility value is ok to leave as "1 - user level".



Now with this done and a "display filter" of "syslog", you should be able to see events start to appear in WireShark such as below. You can also point multiple devices' syslog configuration to the same PC running WireShark and later filter all the collected data by IP address. This may be helpful when you are trying to track 2 different devices and want to be able to match up which event happened in which specific order real time.



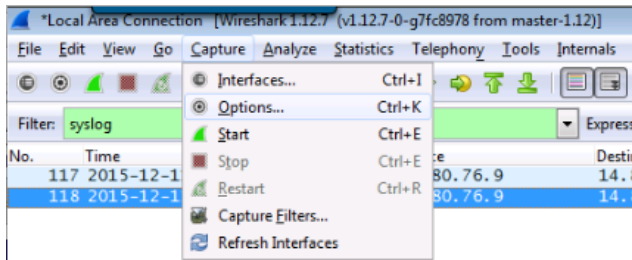
No.	Time	Source	Destination	Protocol	Length	Info
499	2015-12-11 16:13:54.845392000	14.80.76.9	14.80.98.182	Syslog	115	USER.NOTICE: Dec 11 16:20:36 mse8510-76-9 SIP incoming call from 14.80.98.175:5061
500	2015-12-11 16:13:54.846260000	14.80.76.9	14.80.98.182	Syslog	117	USER.NOTICE: Dec 11 16:20:36 mse8510-76-9 CONFERENCE new incoming SIP participant 11
501	2015-12-11 16:13:54.846810000	14.80.76.9	14.80.98.182	Syslog	129	USER.NOTICE: Dec 11 16:20:36 mse8510-76-9 CONFERENCE participant 11 name set to "SX20 ISDN Link"
502	2015-12-11 16:13:54.846964000	14.80.76.9	14.80.98.182	Syslog	133	USER.NOTICE: Dec 11 16:20:36 mse8510-76-9 CONFERENCE participant 11 is joining conference "mlimbric"
503	2015-12-11 16:13:54.850893000	14.80.76.9	14.80.98.182	Syslog	143	USER.NOTICE: Dec 11 16:20:36 mse8510-76-9 CONFERENCE activating h.239 channel (h.263 640 x 480) for "mlimbric"
510	2015-12-11 16:13:55.092196000	14.80.76.9	14.80.98.182	Syslog	146	USER.WARNING: Dec 11 16:20:37 mse8510-76-9 RTP call 11: Processing unencrypted RTP messages for handler 35334
511	2015-12-11 16:13:55.134164000	14.80.76.9	14.80.98.182	Syslog	146	USER.WARNING: Dec 11 16:20:37 mse8510-76-9 RTP call 11: Processing unencrypted RTP messages for handler 35334
681	2015-12-11 16:13:58.619908000	14.80.76.9	14.80.98.182	Syslog	174	USER.WARNING: Dec 11 16:20:40 mse8510-76-9 API unable to handle XML RPC request device.query from 14.80.99.146
1098	2015-12-11 16:14:13.387608000	14.80.76.9	14.80.98.182	Syslog	191	USER.NOTICE: Dec 11 16:20:55 mse8510-76-9 CONFERENCE destroy SIP participant 11, "SX20 ISDN Link". Reason: part
1099	2015-12-11 16:14:13.387611000	14.80.76.9	14.80.98.182	Syslog	136	USER.NOTICE: Dec 11 16:20:55 mse8510-76-9 CONFERENCE removing participant 11 from conference "mlimbric"
1102	2015-12-11 16:14:13.614361000	14.80.76.9	14.80.98.182	Syslog	126	USER.NOTICE: Dec 11 16:20:55 mse8510-76-9 CONFERENCE deactivated h.239 channel for "mlimbric"
2117	2015-12-11 16:14:49.793596000	14.80.76.9	14.80.98.182	Syslog	174	USER.WARNING: Dec 11 16:21:31 mse8510-76-9 API unable to handle XML RPC request device.query from 14.80.99.146
3592	2015-12-11 16:15:41.033580000	14.80.76.9	14.80.98.182	Syslog	174	USER.WARNING: Dec 11 16:22:23 mse8510-76-9 API unable to handle XML RPC request device.query from 14.80.99.146

These are the basic steps to configure WireShark as a Syslog server for any device.

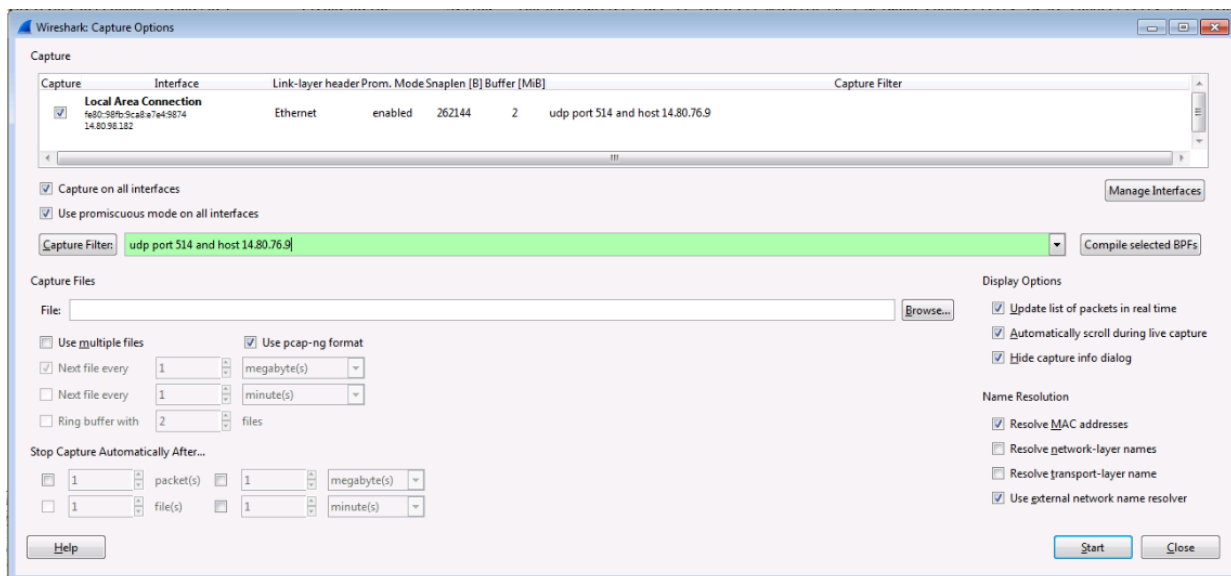
Advanced configurations

On an MCU, the Event Logs only contain the last 1999 lines of logs which overwrite and cycle through with the newest entries. If you need to enable Trace level of debug logging, you may fill up this log in a couple minutes depending on the event you are trying to monitor. This is the benefit of a Syslog server. It will take all the logs you send it and keep them without overwriting. (Please see your specific Syslog server documentation to further validate).

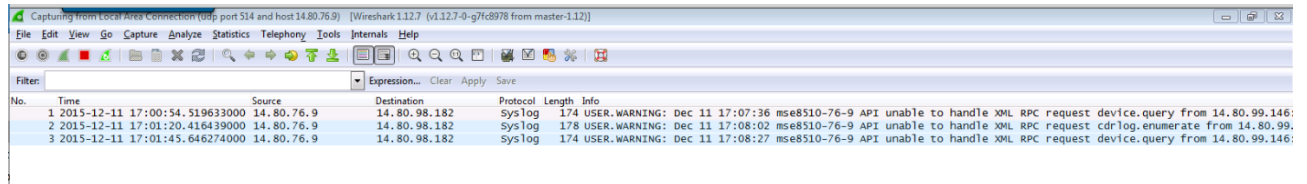
To best make use of WireShark as a Syslog server, there are settings we can configure to minimize packet capture size. Typical syslog messages are UDP using port 514. As such, we can create a "Capture Filter" in WireShark to only capture packets destined for UDP port 514. Do begin, in WireShark, go to (WireShark > Capture > Options)



In the field next to the "Capture Filter" button, type "udp port 514 and host <ip of wireshark>" i.e. 14.80.76.9. If the syntax is correct, the field should highlight green. If so, click Start. If we want to capture syslog event from multiple IP addresses, just use "udp port 514" as a "capture filter". That way you can sort the data later as desired.



Now with the "capture filter" above applied, Wireshark will only capture UDP packets from 14.80.76.9:514. All other packets will be ignored. Notice there is no "Display Filter" needed as we are already filtering which packets to capture. We can now safely run the WireShark capture for several hours without worry of size as we are only capturing a minimal amount of packets.

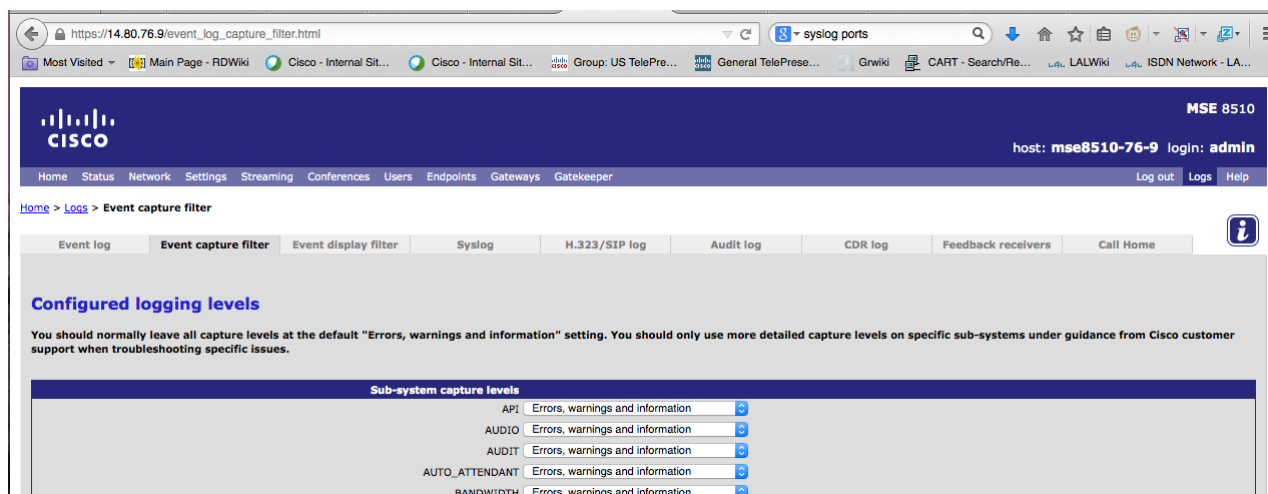


Enable increased MCU logging

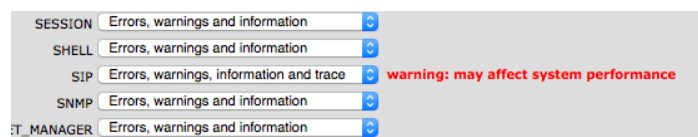
There are times where we will want to run a test for an extended period with increased log levels, for instance a SIP call. Typically the H.323/SIP logs (MCU > Logs > H.323/SIP Logs) when enabled will capture most of the dialog needed (also know as Protocol logs depending on MCU version), but if more detail is needed, we can enable Event Capture Filters for specific issues. This will increase the Event log output, which is what is sent to the Syslog server. H.323/SIP (or Protocol) logs are only stored locally and are not sent to the syslog server. TelePresence MCU's have the ability starting in 4.1(1.79) to transfer the Protocol logs to a HTTP(S) client (not covered in this article). For more details please see the article below: (Cisco Internal)

<https://techzone.cisco.com/t5/Conferencing/How-to-capture-extended-Protocol-Logging-for-TelePresence-Server/ta-p/836459>

When in need to increase debug logging, go to (MCU > Logs > Event Capture Filters).



By default, all logging is set to "Errors, warnings and information" (EWI). This is the log level that should be set for normal use. When in need to increase logging, you will want to select "Errors, warnings, information and trace" (EWIT). Once you have made your selection, click "Update Settings". You will be prompted to confirm your selection as this may effect performance. Ideally you would want to run these log levels when there is low system utilization. As you see below, I have enabled SIP debug levels to (EWIT) and have a warning next to it (after updating the settings) to inform this may affect system performance. Keep in mind if there is an issue, there is usually good reason we need to run these traces.



Note: Please be sure to set the log levels back to (EWI) once done testing.

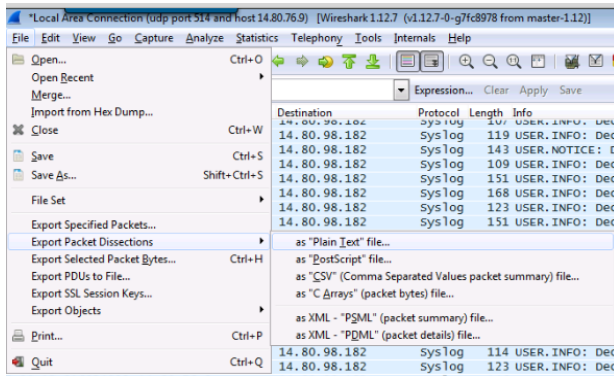
Now looking at Wireshark, we will have a lot more activity, especially when running another SIP call as before. As you can see below, a lot more details are logged on the MCU. Keep in mind this is not a port span of the MCU, so you will not see the entire SIP dialog messages as you will in the Protocol logs, but this will display how the MCU processes these messages internally. If you want to capture the SIP dialog messages, use the Protocol logs or a port span.

Source	Destination	Protocol	Length	Info
7:30:11.687053000	14.80.76.9	14.80.98.182	117	USER.INFO: Dec 11 17:36:53 mse8510-76-9 SIP Starting new INVITE server transaction
7:30:11.687056000	14.80.76.9	14.80.98.182	131	USER.INFO: Dec 11 17:36:53 mse8510-76-9 SIP Starting invite server transaction - transport = TLS
7:30:11.687057000	14.80.76.9	14.80.98.182	102	USER.INFO: Dec 11 17:36:53 mse8510-76-9 SIP Handling INVITE request
7:30:11.687079000	14.80.76.9	14.80.98.182	179	USER.INFO: Dec 11 17:36:53 mse8510-76-9 SIP Extracting URI from "<sip:sx20@l1mbric.tmspe.labs;pr=urn:uuid:9713d447-7e34-558c-b84
7:30:11.687680000	14.80.76.9	14.80.98.182	168	USER.INFO: Dec 11 17:36:53 mse8510-76-9 SIP Extracting URI from ""sx20 ISDN Link"<sip:sx20@l1mbric.tmspe.labs;tag=f8d7ffc4f8c
7:30:11.687681000	14.80.76.9	14.80.98.182	130	USER.INFO: Dec 11 17:36:53 mse8510-76-9 SIP Extracting URI from "<sip:6650@l1mbric.tmspe.labs"
7:30:11.687682000	14.80.76.9	14.80.98.182	130	USER.INFO: Dec 11 17:36:53 mse8510-76-9 SIP Extracting URI from "<sip:6650@l1mbric.tmspe.labs"
7:30:11.687683000	14.80.76.9	14.80.98.182	147	USER.INFO: Dec 11 17:36:53 mse8510-76-9 SIP Extracting URI from ""sx20 ISDN Link"<sip:sx20@l1mbric.tmspe.labs"
7:30:11.687684000	14.80.76.9	14.80.98.182	133	USER.INFO: Dec 11 17:36:53 mse8510-76-9 SIP Registration handle 40020002 found in incoming request
7:30:11.687684000	14.80.76.9	14.80.98.182	284	USER.INFO: Dec 11 17:36:53 mse8510-76-9 SIP Extracting URI from "<sip:proxy-call-id=5dfea311-d3c0-4d50-8644-9d70b3638c78014.80.9
7:30:11.687685000	14.80.76.9	14.80.98.182	117	USER.INFO: Dec 11 17:36:53 mse8510-76-9 SIP Sending response 100 Trying no content
7:30:11.687686000	14.80.76.9	14.80.98.182	130	USER.INFO: Dec 11 17:36:53 mse8510-76-9 SIP Extracting URI from "<sip:6650@l1mbric.tmspe.labs"
7:30:11.688184000	14.80.76.9	14.80.98.182	113	USER.INFO: Dec 11 17:36:53 mse8510-76-9 SIP Sent 100 Trying to 14.80.98.175:5061
7:30:11.688186000	14.80.76.9	14.80.98.182	101	USER.INFO: Dec 11 17:36:53 mse8510-76-9 SIP Handling incoming call
7:30:11.688187000	14.80.76.9	14.80.98.182	127	USER.INFO: Dec 11 17:36:53 mse8510-76-9 SIP Found existing TCP handle for 14.80.98.175:5061
7:30:11.688187000	14.80.76.9	14.80.98.182	118	USER.INFO: Dec 11 17:36:53 mse8510-76-9 SIP Route 2 for c00b0000 linked to 00001e8e
7:30:11.688188000	14.80.76.9	14.80.98.182	92	USER.INFO: Dec 11 17:36:53 mse8510-76-9 SIP Analysing SDP
7:30:11.689130000	14.80.76.9	14.80.98.182	117	USER.NOTICE: Dec 11 17:36:53 mse8510-76-9 CONFERENCE new incoming SIP participant 12
7:30:11.689104000	14.80.76.9	14.80.98.182	129	USER.INFO: Dec 11 17:36:53 mse8510-76-9 SIP Matching: name = sx20 ISDN Link, IP = 14.80.96.149
7:30:11.689105000	14.80.76.9	14.80.98.182	129	USER.NOTICE: Dec 11 17:36:53 mse8510-76-9 CONFERENCE participant 12 name set to "sx20 ISDN Link"
7:30:11.689352000	14.80.76.9	14.80.98.182	102	USER.INFO: Dec 11 17:36:53 mse8510-76-9 SIP New connection accepted
7:30:11.689354000	14.80.76.9	14.80.98.182	113	USER.INFO: Dec 11 17:36:53 mse8510-76-9 SIP Have local signalling ip = 14.80.76.9
7:30:11.689354000	14.80.76.9	14.80.98.182	96	USER.INFO: Dec 11 17:36:53 mse8510-76-9 SIP MCU not ready yet
7:30:11.689361000	14.80.76.9	14.80.98.182	177	USER.INFO: Dec 11 17:36:53 mse8510-76-9 SIP CC has received updated remote identity information "sx20 ISDN Link"<sip:sx20@l1mb

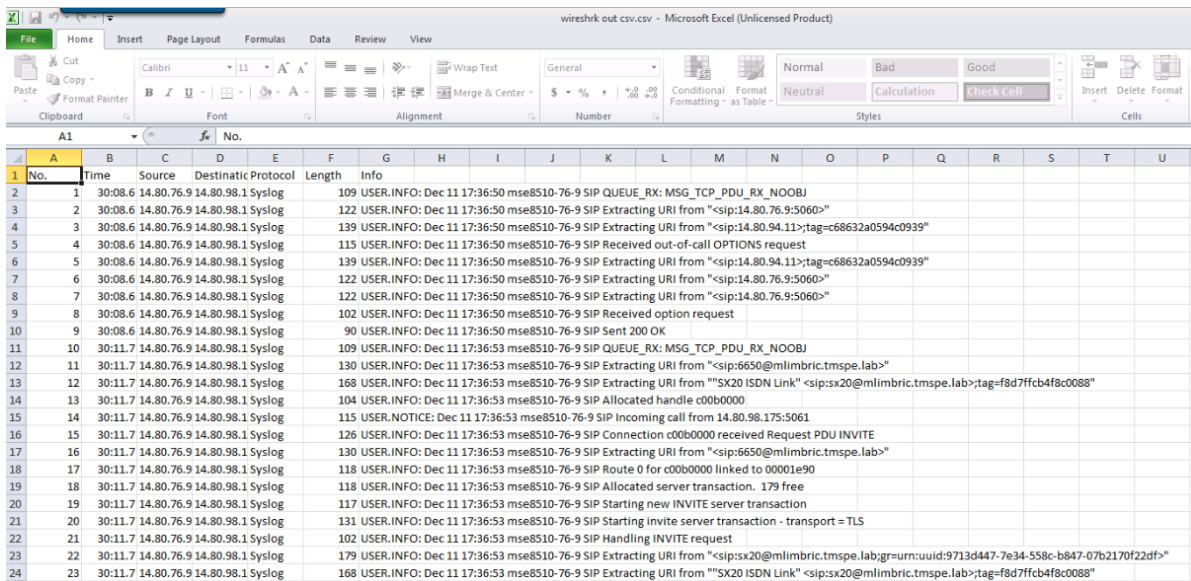
Saving the WireShark data

After you stop the capture, having collected the data you were looking for in WireShark from the syslogs, you can export as a new packet capture or text file. In WireShark, go to (File > Export Specified Packets...). This is typically used when you have a large capture and you want to export currently showing packets displayed on screen from a specific "Display Filter". If you have only a "Capture Filter" enabled, you can save the WireShark capture as a normal pcap file (File > Save As...).

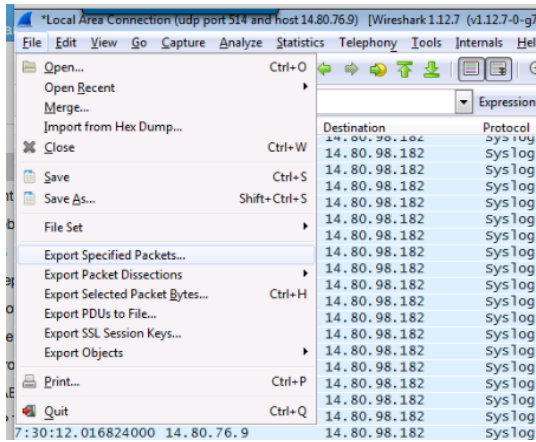
To Export to a text file, I recommend (File > Export Packet Dissections > as "CSV" (Comma Separated Values packet summary) file...) format



I find it is easier to look at and closely resembles the WireShark display output when viewing with Excel.

A screenshot of a Microsoft Excel spreadsheet. The spreadsheet contains data exported from Wireshark in CSV format. The columns are labeled: No., Time, Source, Destination, Protocol, Length, and Info. The data consists of 24 rows of network packet information, including timestamps, IP addresses, protocols (Syslog), and detailed packet descriptions such as 'USER.INFO: Dec 11 17:36:50 mse8510-76-9 SIP QUEUE_RX: MSG_TCP_PDU_RX_NOOBJ' and 'SIP Extracting URI from <\"/>

Alternatively, to export the capture as a text file, go to (File > Export Packet Dissections > as "Plain Text" file...).



This view is not formatted as clean as the CSV format

