

OneM2M Dissector for Wireshark

Contents

1. Overview	3
2. Installing the Dissector	3
3. Dissector and General Capabilities	3
4. Installing the Dissector	3
4.1. Configuring the Dissector	3
5. Using the Dissector	4
5.1. Launching the Dissector	4
5.2. Display Filter	6
5.3. oneM2M Message Dissection	7
5.4. Coloring Rules	9

1. Overview

The InterDigital OneM2M Dissector for Wireshark analyzes and display oneM2M requests and responses. It is developed to display the packet info according to OneM2M structure. The current version supports oneM2M R1 bindings for both HTTP and CoAP. This document describes how to configure and use the OneM2M dissector for Wireshark.

2. Installing the Dissector

The OneM2M dissector is used as a plugin in Wireshark. It can be installed on any machine that has Wireshark **v1.12.0** (or later) installed.

3. Dissector and General Capabilities

Wireshark gives the ability to store and view internal logs and proprietary protocols by writing custom dissectors. With the help of OneM2M dissector, one can view live communication or analyze the offline logs without additional formatting. With OneM2M dissector, one can view the decoded OneM2M protocol in user readable structure and format.

4. Installing the Dissector

This section describes how to configure the dissector in Wireshark.

The dissector consists of two LUA files (onem2m.lua and coap18.lua). To install the dissector, the lua files need to be copied into the Wireshark plugins folder. The location of plugins folder for Mac, Windows and Linux environments is given below:

- **Mac:** `~/wireshark/plugins` (create plugins folder if it doesn't exist)
- **Windows:** `"C:\Program Files\Wireshark\plugins\1.12.0"` (replace 1.12.0 with your Wireshark version number)
- **Linux:** `~/wireshark/plugins` (create plugins folder if it doesn't exist)

Once installed, make sure that `disable_lua = false` in `init.lua`

4.1. Configuring the Dissector

This section describes how to configure specific ports of the dissector so that oneM2M messages are automatically decoded on these ports.

At the end of the lua file, a **"tcp_table"** dissector table exists for defining http ports and a **"udp_table"** dissector table exists for defining CoAP ports. Adding port numbers to the dissector table enables automatic dissection of oneM2M messages occurring over these specific ports.

```

--register_postdissector(onem2mProto)
--load the tcp.port table
tcp_table = DissectorTable.get("tcp.port")
-- register ONEM2M protocol to handle tcp ports
tcp_table:add(56983,onem2mProto)
tcp_table:add(6060,onem2mProto)
tcp_table:add(50386,onem2mProto)
tcp_table:add(6000,onem2mProto)
tcp_table:add(61745,onem2mProto)
tcp_table:add(8011,onem2mProto)
tcp_table:add(9011,onem2mProto)

-- load the udp.port table
udp_table = DissectorTable.get("udp.port")
-- register ONEM2M protocol to handle udp ports
udp_table:add(10062, onem2mProto)
udp_table:add(8010,onem2mProto)
udp_table:add(61617,onem2mProto)
udp_table:add(9017,onem2mProto)
udp_table:add(61616,onem2mProto)
udp_table:add(9010,onem2mProto)

```

Add new http port number
using tcp_table:add(xxxx,
onem2mProto)

Add new CoAP port number
using udp_table:add(xxxx,
onem2mProto)

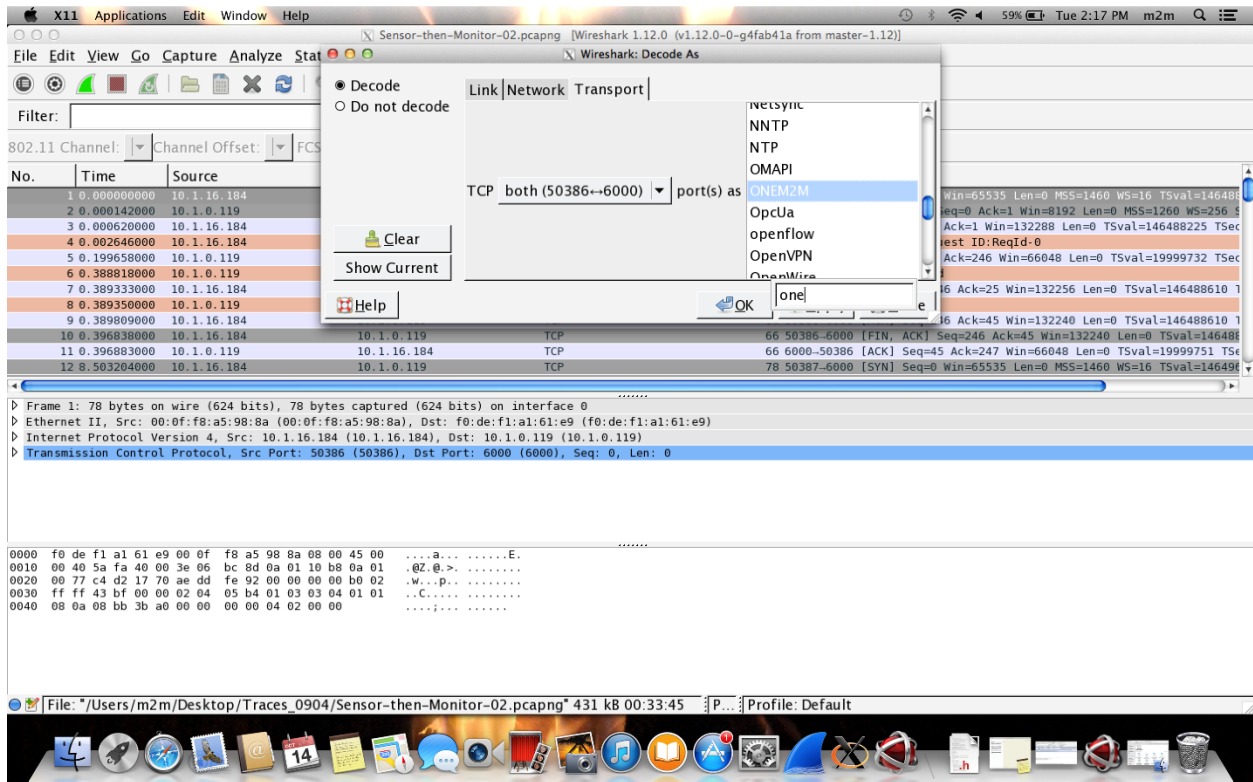
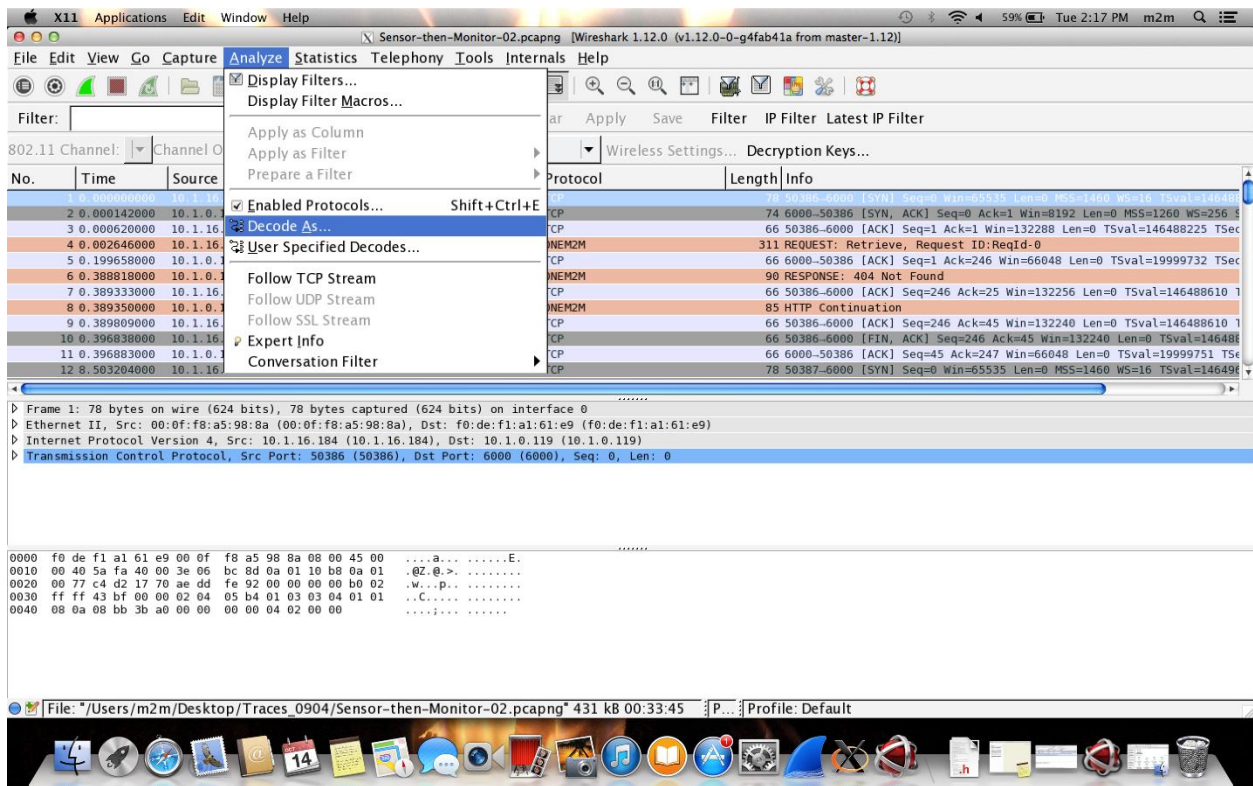
5. Using the Dissector

This section describes how to run the dissector.

5.1. Launching the Dissector

To start the dissector, simply open a previous Wireshark capture file or start a new capture in Wireshark. If a packet's source and destination ports are present in the dissector, then that packet will automatically be decoded as a oneM2M packet.

If the packet's source and destination ports are not present in the dissector table, then one can use the "Decode As" feature of Wireshark to select the oneM2M protocol option. To dissect explicitly, go to menu bar, select Analyze -> Decode As, select "OneM2M" from the drop down.



5.2. Display Filter

To show only OneM2M messages enter “onem2m” in the Wireshark “Filter” dialogue box

The screenshot shows the Wireshark interface with the filter 'onem2m' applied. The packet list displays several OneM2M messages. The detailed view shows the structure of a OneM2M message, including the Ethernet II, Internet Protocol Version 4, Transmission Control Protocol, and Hypertext Transfer Protocol layers. The OneM2M Protocol Data section shows the message structure, including the request ID and the payload.

No.	Time	Source	Destination	Protocol	Length	Info
4	0.001552000	10.1.0.181	10.1.5.25	ONEM2M	267	REQUEST: Create, REQUEST ID:ASN-CSE_34
6	0.204827000	10.1.0.181	10.1.5.25	ONEM2M/XML	423	OneM2M Primitive Payload
8	0.620299000	10.1.5.25	10.1.0.181	ONEM2M	76	RESPONSE: 201 Created
9	0.620374000	10.1.5.25	10.1.0.181	ONEM2M	125	HTTP Continuation
16	1.245309000	10.1.0.181	10.1.5.25	ONEM2M	174	REQUEST: Retrieve, REQUEST ID:ASN-CSE_41
18	1.828327000	10.1.5.25	10.1.0.181	ONEM2M	76	RESPONSE: 200 Content
19	1.828643000	10.1.5.25	10.1.0.181	ONEM2M/XML	669	HTTP Continuation
28	43.343318000	10.1.0.181	10.1.5.25	ONEM2M/XML	194	OneM2M Primitive Payload
29	43.455753000	10.1.5.25	10.1.0.181	ONEM2M	80	RESPONSE: 400 Bad Request
30	43.455859000	10.1.5.25	10.1.0.181	ONEM2M	164	HTTP Continuation

Frame 26: 289 bytes on wire (2312 bits), 289 bytes captured (2312 bits) on interface 0

Ethernet II, Src: f0:de:f1:a1:61:e9 (f0:de:f1:a1:61:e9), Dst: 00:0f:f8:a5:98:8a (00:0f:f8:a5:98:8a)

Internet Protocol Version 4, Src: 10.1.0.181 (10.1.0.181), Dst: 10.1.5.25 (10.1.5.25)

Transmission Control Protocol, Src Port: 51229 (51229), Dst Port: 9011 (9011), Seq: 1, Ack: 1, Len: 235

Hypertext Transfer Protocol

ONEM2M Protocol Data

0000 00 0f f8 a5 98 8a f0 de f1 a1 61 e9 08 00 45 00a...E.
0010 01 13 59 4a 40 00 80 06 86 cb 0a 01 00 b5 0a 01 ..YJ0...
0020 05 19 c8 1d 23 33 0c 56 c7 3d 1a 05 10 9e 50 18#3.V...P.
0030 01 04 58 ef 00 00 50 4f 53 54 20 2f 49 4e 2d 43 ..X...P0 ST /IN-C
0040 53 45 3f 20 48 54 54 50 2f 31 2e 31 0d 0a 48 6f SE? HTTP /1.1..Ho
0050 73 74 3a 20 31 30 2a 31 2e 35 2e 32 35 3a 39 30 st: 10.1.5.25;90
0060 31 31 0d 0a 41 63 63 65 70 74 3a 20 2a 2f 2a 0d 11.Acce pt: */*.
0070 0a 43 6f 6e 74 65 6e 74 2d 54 79 70 65 3a 20 61 .Content -Type: a
0080 70 70 6c 69 63 61 74 69 6f 6e 2f 6f 6e 65 6d 32 pplicati on/onem2

File: /Users/m2m/Desktop/Traces_0904/1015/IN-CSE-01.pcapng* 5200 bytes 00:00:43 P... Profile: Default

5.3. oneM2M Message Dissection

The image shows a Wireshark packet capture of a oneM2M message. The packet list at the top shows a GET request from 10.1.17.3 to 10.1.1.182. The packet details pane shows the HTTP header and the ONEM2M protocol data.

Info displaying request type and request id

oneM2M Request Parameters shown in HTTP Header

oneM2M Request

Frame 4: 206 bytes on wire (1648 bits), 206 bytes captured (1648 bits) on interface 0
Ethernet II, Src: Cisco_a5:98:8a (00:0f:f8:a5:98:8a), Dst: wistronI_a1:61:e9 (f0:de:f1:a1:61:e9)
Internet Protocol Version 4, Src: 10.1.17.3 (10.1.17.3), Dst: 10.1.1.182 (10.1.1.182)
Transmission Control Protocol, Src Port: 49378 (49378), Dst Port: 9011 (9011), Seq: 1, Ack: 1, Len: 140
Hypertext Transfer Protocol
GET /IN-CSE HTTP/1.1\r\nUser-Agent: curl/7.37.1\r\nHost: 10.1.1.182:9011\r\nAccept: */*\r\nX-M2M-RI: xyz01\r\nX-M2M-Origin: http://abc:0000/IN-CSE\r\n\r\n[Full request URI: http://10.1.1.182:9011/IN-CSE]
[HTTP request 1/1]
[Response in frame: 5]
ONEM2M Protocol Data
Request-Line
Operation: Retrieve
To: /IN-CSE HTTP/1.1
Filter Criteria
Header
From: http://abc:0000/IN-CSE
Request Identifier: xyz01

CSEBase.pcapng [Wireshark 1.12.5 (v1.12.5-0-g5819e5b from master-1.12)]

File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help

Filter: Expression... Clear Apply Save

No.	Time	Source	Destination	Protocol	Length	Info
4	0.001429000	10.1.17.3	10.1.1.182	ONEM2M	206	REQUEST: RETRIEVE, REQUEST ID: xyz01
5	0.151859000	10.1.1.182	10.1.17.3	HTTP	88	HTTP/1.1 200 Content
6	0.152162000	10.1.1.182	10.1.17.3	ONEM2M/XML	620	RESPONSE: 2000 OK

Frame 6: 620 bytes on wire (4960 bits), 620 bytes captured (4960 bits) on interface 0

- Ethernet II, Src: wistronI_al:61:e9 (f0:de:f1:a1:61:e9), Dst: Cisco_a5:98:8a (00:0f:f8:a5:98:8a)
- Internet Protocol Version 4, Src: 10.1.1.182 (10.1.1.182), Dst: 10.1.17.3 (10.1.17.3)
- Transmission Control Protocol, Src Port: 9011 (9011), Dst Port: 49378 (49378), Seq: 23, Ack: 141, Len: 554
- Hypertext Transfer Protocol
 - Content-Type: application/vnd.onem2m-res+xml\r\n
 - X-M2M-RI: xyz01\r\n
 - X-M2M-RSC: 2000\r\n
 - Content-Length: 450 \r\n
- Media Type
- ONEM2M Protocol Data
 - Status-Line
 - Response Status Code: 2000 OK
 - Header
 - Request Identifier: xyz01
 - Content
 - extensible Markup Language
 - <?xml
 - <m2m:csb
 - xmlns:m2m="http://www.onem2m.org/xml/protocols"
 - xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 - xsi:schemaLocation="http://www.onem2m.org/xml/protocols CDT-csb-v1_0_0.xsd"
 - name="csb">
 - <ty>5</ty>
 - <ct>20150713T100026</ct>
 - <lt>20150713T100026</lt>
 - <lbl>IN-CSE</lbl>
 - <acpi>/IN-CSE/IN-CSEAcpi</acpi>
 - <cst>1</cst>
 - <csi>IN-CSE</csi>

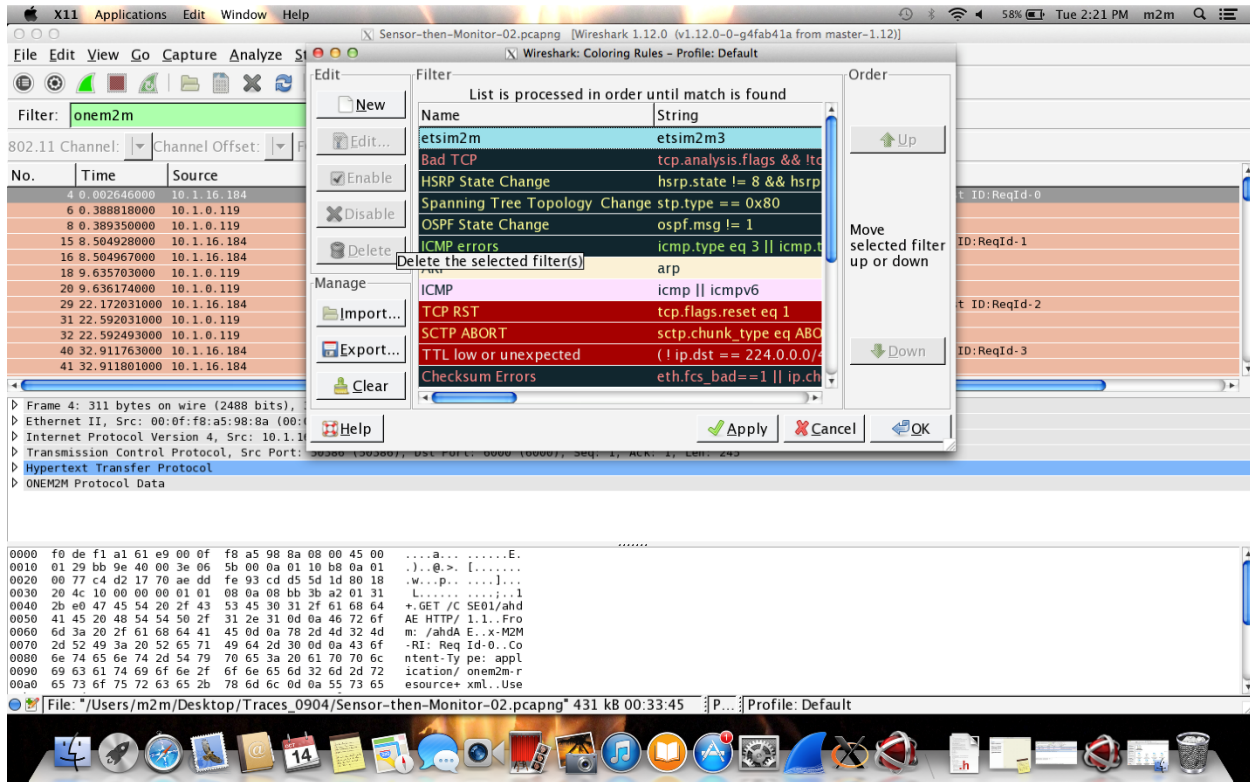
oneM2M Response

0000 00 0f f8 a5 98 8a f0 de f1 a1 61 e9 08 00 45 00a...E.
0010 02 5e 6b c7 40 00 80 06 66 18 0a 01 01 b6 0a 01 ..^k.@... f.....
0020 11 03 23 33 c0 e2 14 5b b0 34 88 0e 66 41 80 19 ..#3...[.4..fA..
0030 01 02 a0 82 00 00 01 01 08 0a 01 6c 6c 03 3b a5ll.:.
0040 f6 c4 43 6f 6e 74 65 6e 74 2d 54 79 70 65 3a 20 ..Content t-Type:
0050 61 70 70 6c 60 62 61 74 60 6f 60 7f 76 60 6d 70 application/vnd

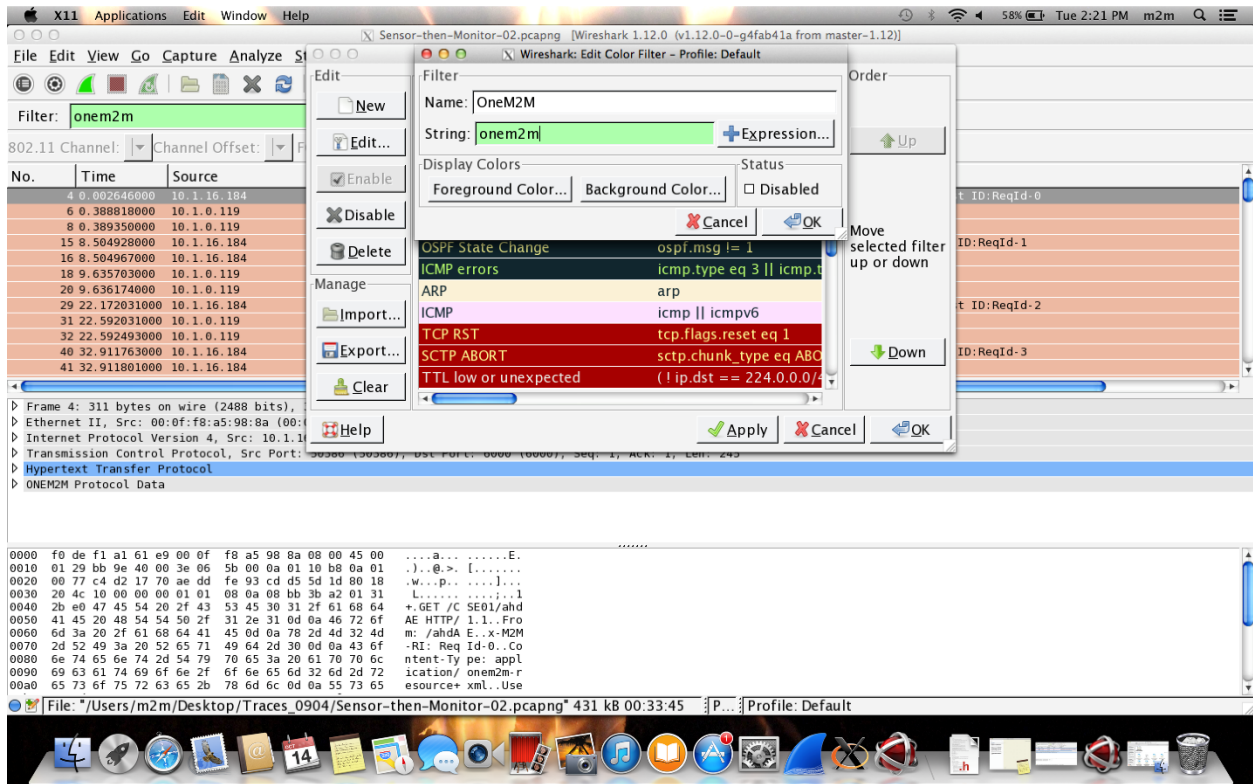
File: "C:\Users\seeddn\Documents\Temp\W... Packets: 330 · Displayed: 330 (100.0%) · Load time: 0:00.020 Profile: Default

To distinguish OneM2M protocol from others, one can set the colors for OneM2M packets. To do this,

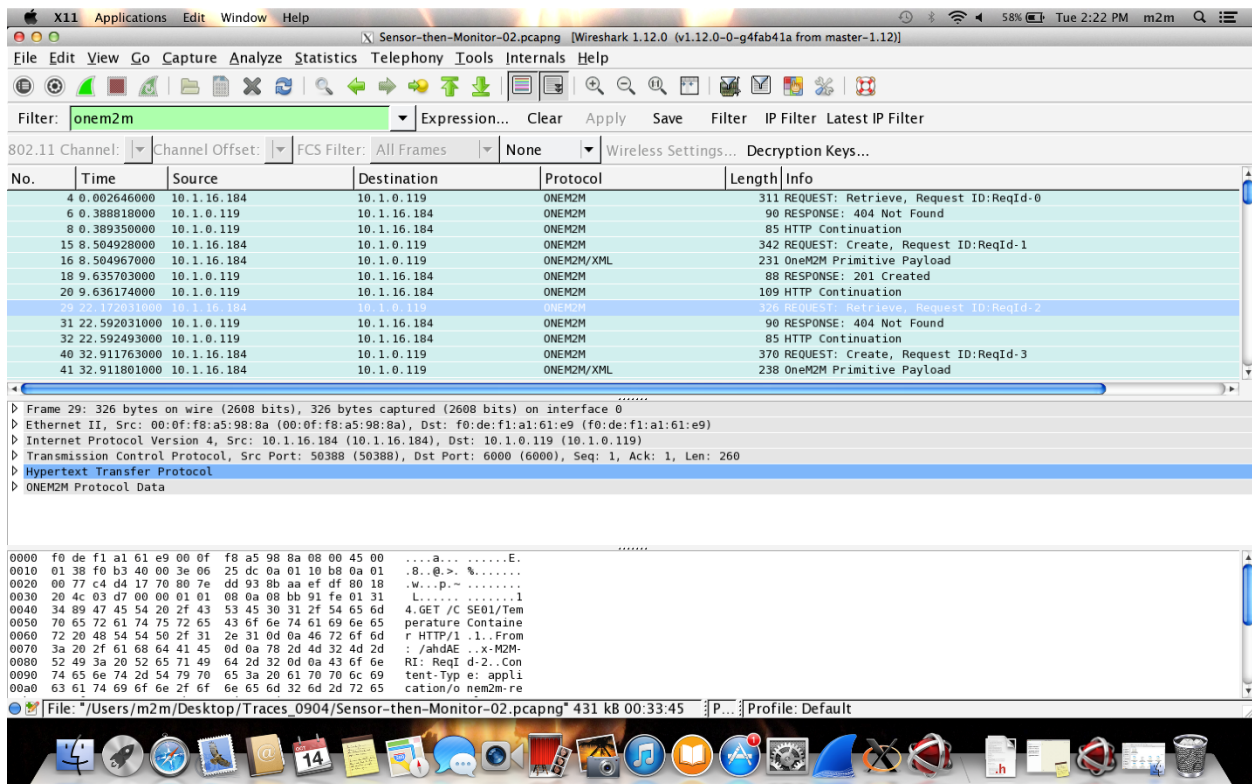
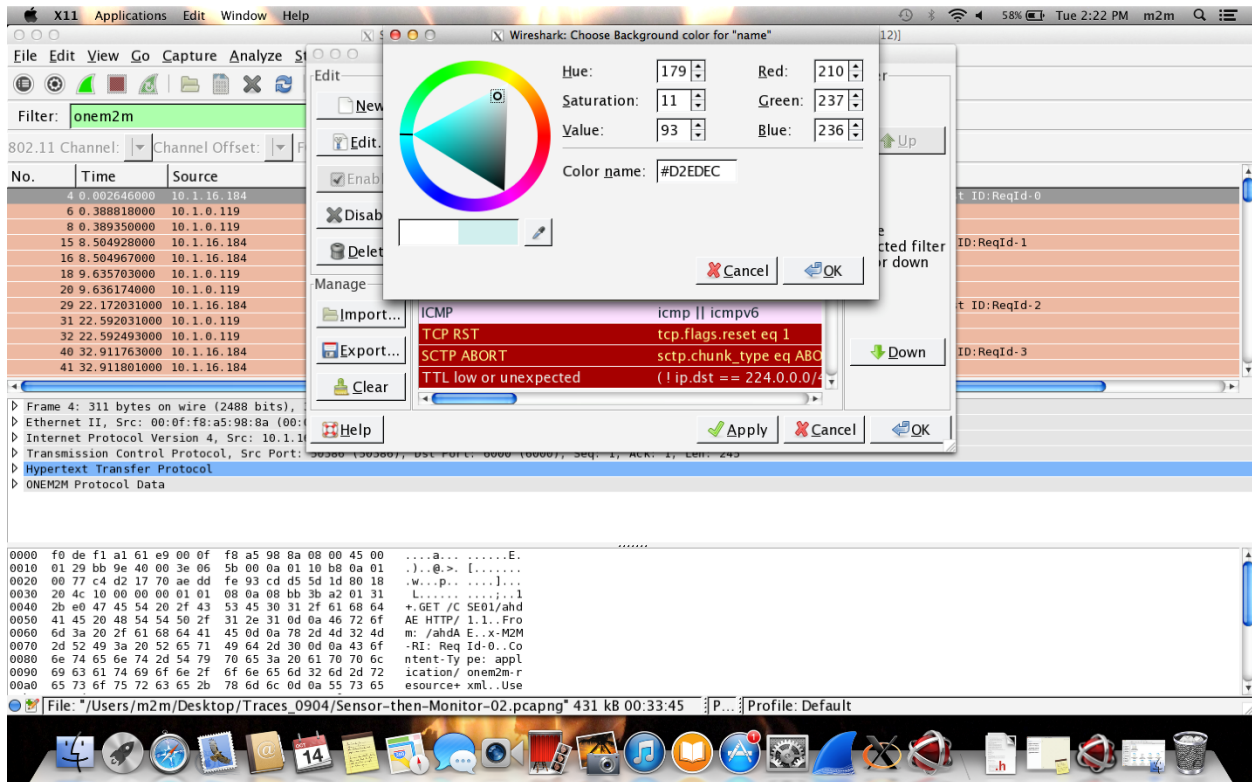
1. In the menu bar, go to View-> Coloring Rules



2. Give a name to the coloring rule, write "OneM2M" in string field.



3. Choose foreground and background colors, press "OK"



About InterDigital®

InterDigital develops fundamental wireless technologies that are at the core of mobile devices, networks, and services worldwide. As a long-standing contributor to the evolution of the wireless industry, we solve many of the industry's most critical and complex technical challenges years ahead of market deployment. Our advanced solutions support more efficient wireless networks, a richer multimedia experience, and new mobile broadband capabilities. Accordingly, we have established licenses and strategic relationships with many of the world's leading wireless companies.

InterDigital, Inc.
200 Bellevue Parkway
Suite #300
Wilmington, DE 19809-3727
www.interdigital.com

© InterDigital, Inc. 2014. All rights reserved. This work was prepared and contains information supplied by, InterDigital, Inc. and/or its affiliates (hereinafter, "InterDigital"). All information, including performance information, contained herein is provided on an AS IS basis without any warranty as to its accuracy or results. InterDigital expressly disclaims any and all liability for any errors or omissions. InterDigital reserves the right to modify this work and the information contained herein without notice. No part of this work may be reproduced, in whole or in part, except as authorized in writing by InterDigital, irrespective of the type of media in which the information may be embodied. "InterDigital" is a registered trademark of InterDigital, Inc.