Kepware Technologies

Using Wireshark for Ethernet Diagnostics

March, 2012
Ref. 50.08

Table of Contents

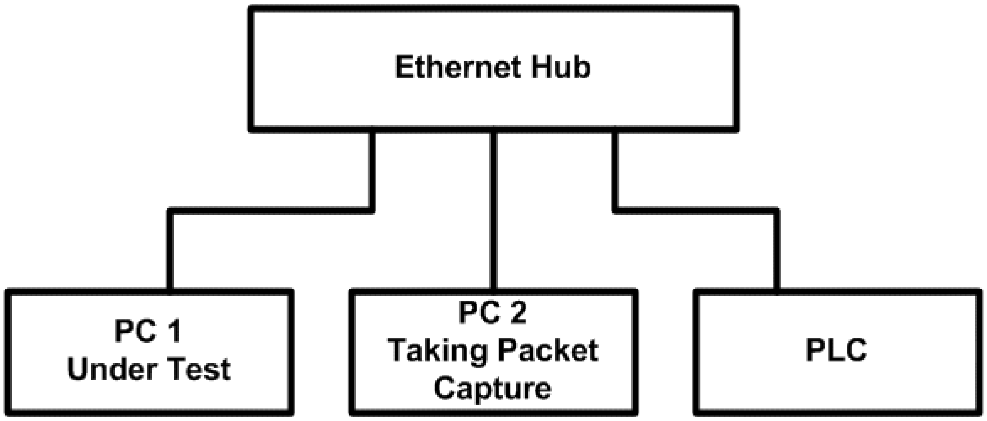
1. Introduction

This document intends to discuss how to get information on the PC's Ethernet communications for troubleshooting. It will demonstrate how to use a software utility like Wireshark to capture all packets being sent or received by the PC's Ethernet card.

2. Setting up the Software

The following examples use packet-capturing software that can run on the following:

1. A PC that requires troubleshooting (such as a "PC under test").
2. A PC that can receive packets being sent to and from the PCs and devices under test. For example, the diagram below displays how to record communications between a PC and a PLC.



A packet sent to an Ethernet Hub (such as from PC 1) is broadcast to all of the hub's ports (such as to PC 2 and the PLC). This allows the packet-capturing software located on PC 2 to monitor the packets being sent by the other PCs and devices (such as PC 1 and the PLC).

In this example, an Ethernet switch could most likely not be used instead of an Ethernet Hub. A packet sent to a switch (such as from PC 1) is only sent to the PC or device to which the packet is addressed (such as to the PLC). If a switch is used, PC 2 will not be able to monitor the traffic between PC 1 and the PLC. Although some switches allow one port to monitor another port, the hub always allows traffic to be monitored.

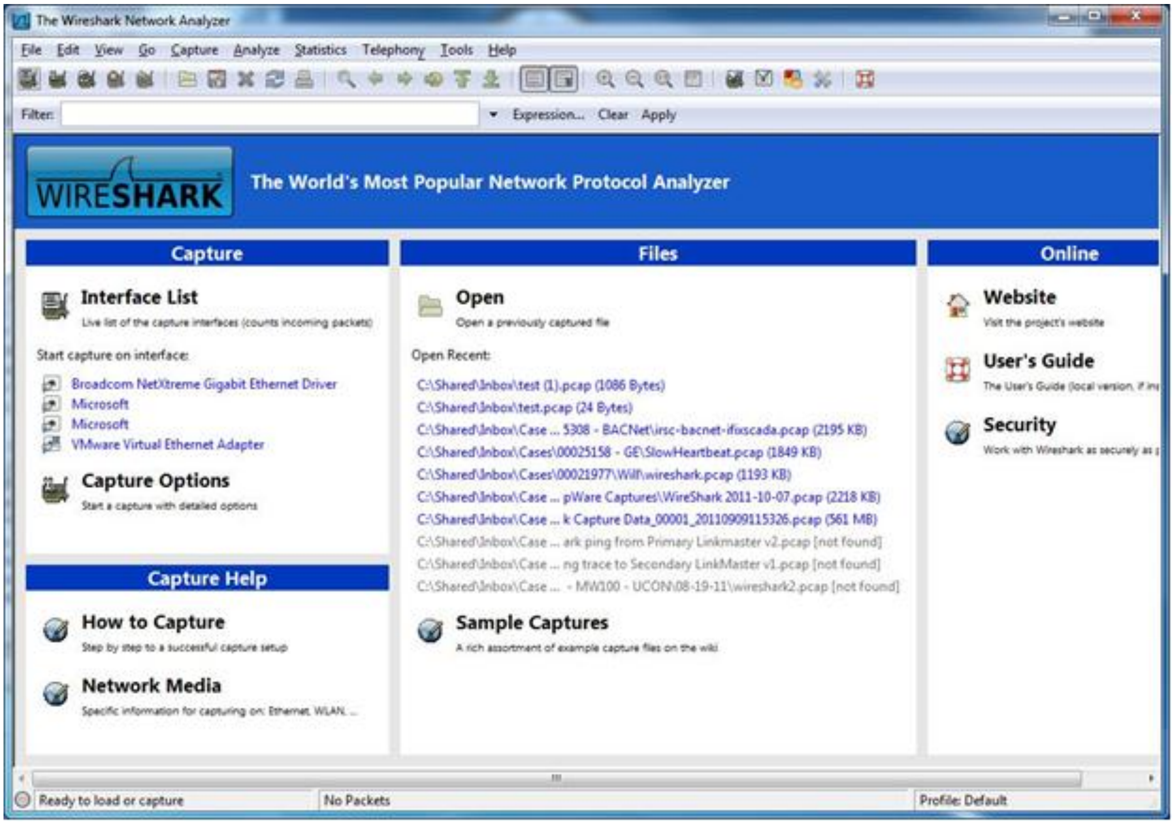
Once users have a PC that can monitor packets being sent to and from the devices under test, a packet-capturing program can be installed on the PC and a capture can be taken.

Note: The following examples use version 1.4.1 of the freeware packet-capture utility Wireshark. This software is available at www.Wireshark.org.

3. Using Wireshark

For information on using the packet-capturing program Wireshark, refer to the instructions below.

1. To start, open **Wireshark**. If this is the first time it is being opened, click **Edit** | **Preferences**.



The World's Most Popular Network Protocol Analyzer

Capture

Interface List
Live list of the capture interfaces (counts incoming packets)

Start capture on interface:

- Broadcom NetXtreme Gigabit Ethernet Driver
- Microsoft
- Microsoft
- VMware Virtual Ethernet Adapter

Capture Options
Start a capture with detailed options

Capture Help

How to Capture
Step by step to a successful capture setup

Network Media
Specific information for capturing on: Ethernet, WLAN, ...

Files

Open
Open a previously captured file

Open Recent:

- C:\Shared\Inbox\test (1).pcap (1086 Bytes)
- C:\Shared\Inbox\test.pcap (24 Bytes)
- C:\Shared\Inbox\Case ... 5308 - BACNet\irsc-bacnet-ifixscada.pcap (2195 KB)
- C:\Shared\Inbox\Cases\00025158 - GE\SlowHeartbeat.pcap (1849 KB)
- C:\Shared\Inbox\Cases\00021977\Will\wireshark.pcap (1193 KB)
- C:\Shared\Inbox\Case ... pWare Captures\WireShark 2011-10-07.pcap (2218 KB)
- C:\Shared\Inbox\Case ... k Capture Data_00001_20110909115326.pcap (561 MB)
- C:\Shared\Inbox\Case ... ark ping from Primary Linkmaster v2.pcap [not found]
- C:\Shared\Inbox\Case ... ng trace to Secondary LinkMaster v1.pcap [not found]
- C:\Shared\Inbox\Case ... - MW100 - UCON\08-19-11\wireshark2.pcap [not found]

Sample Captures
A rich assortment of example capture files on the wiki

Online

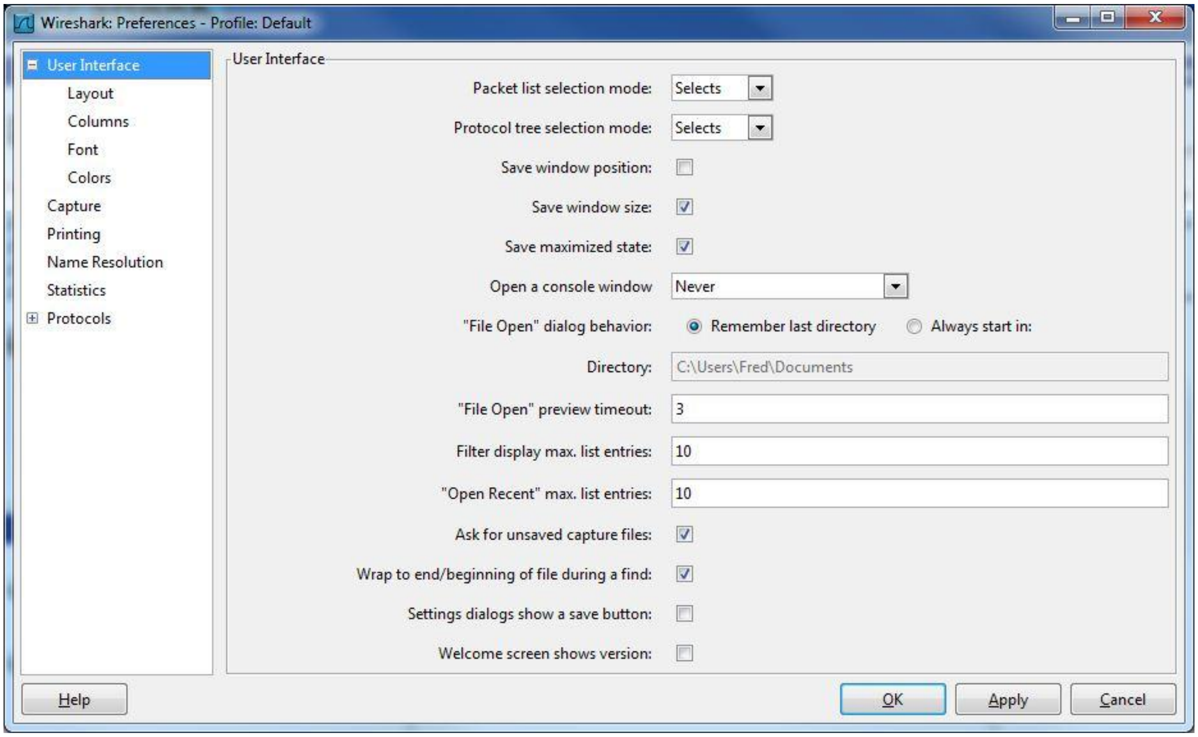
Website
Visit the project's website

User's Guide
The User's Guide (local version, if ins

Security
Work with Wireshark as securely as p

Ready to load or capture | No Packets | Profile: Default

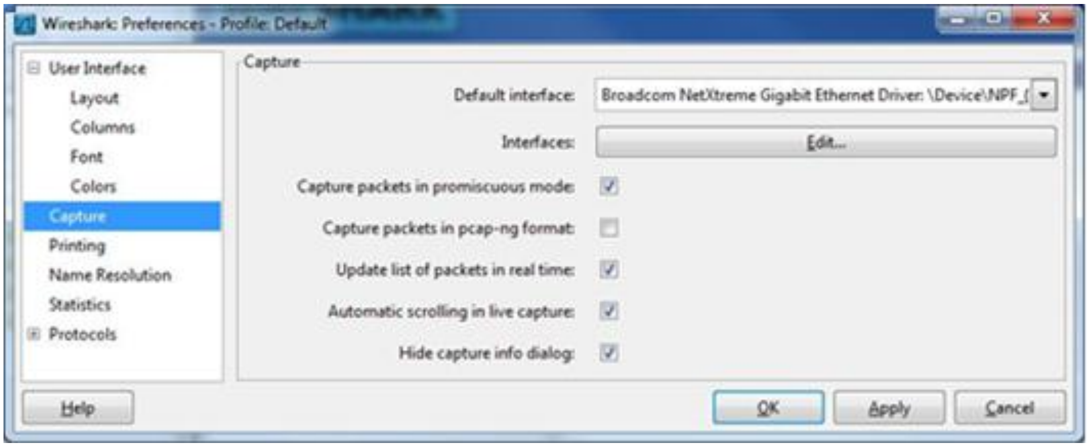
2. In the left-hand pane, select **Capture**.



User Interface

Packet list selection mode: Selects
Protocol tree selection mode: Selects
Save window position: □
Save window size: ☑
Save maximized state: ☑
Open a console window: Never
"File Open" dialog behavior: ● Remember last directory ○ Always start in:
Directory: C:\Users\Fred\Documents
"File Open" preview timeout: 3
Filter display max. list entries: 10
"Open Recent" max. list entries: 10
Ask for unsaved capture files: ☑
Wrap to end/beginning of file during a find: ☑
Settings dialogs show a save button: □
Welcome screen shows version: □

3. In **Default Interface**, select the Ethernet adapter from the drop-down menu.



Capture

Default interface: Broadcom NetXtreme Gigabit Ethernet Driver: \Device\NPF_{

Interfaces: Edit...

Capture packets in promiscuous mode: ☑

Capture packets in pcap-ng format: □

Update list of packets in real time: ☑

Automatic scrolling in live capture: ☑

Hide capture info dialog: ☑

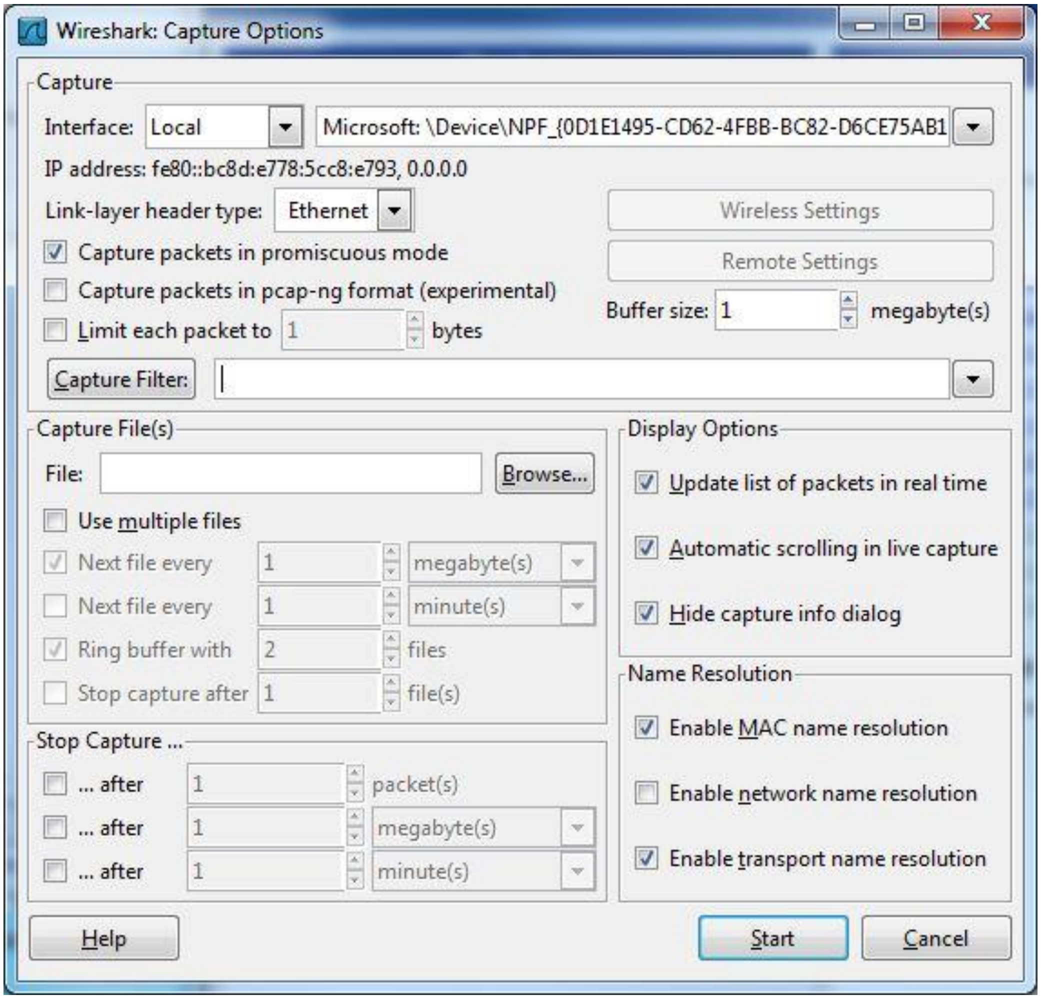
Note: Users must select an adapter the first time that Wireshark is used, regardless of whether there is only one available in the packet-capturing PC.

4. Once finished, click **OK**.

3.1 Selecting Capture Options

The Capture Options specify what network packets will be collected and how the capture files will be handled.

1. To set the default capture options, select **Capture Options** from the main form. Alternatively, click **Capture | Options**.



Capture

Interface: Local | Microsoft: \Device\NPF_{0D1E1495-CD62-4FBB-BC82-D6CE75AB1

IP address: fe80::bc8d:e778:5cc8:e793, 0.0.0.0

Link-layer header type: Ethernet

☑ Capture packets in promiscuous mode

□ Capture packets in pcap-ng format (experimental)

□ Limit each packet to 1 bytes

Buffer size: 1 megabyte(s)

Capture File(s)

File:

□ Use multiple files

☑ Next file every 1 megabyte(s)

□ Next file every 1 minute(s)

☑ Ring buffer with 2 files

□ Stop capture after 1 file(s)

Stop Capture ...

□ ... after 1 packet(s)

□ ... after 1 megabyte(s)

□ ... after 1 minute(s)

Display Options

☑ Update list of packets in real time

☑ Automatic scrolling in live capture

☑ Hide capture info dialog

Name Resolution

☑ Enable MAC name resolution

□ Enable network name resolution

☑ Enable transport name resolution

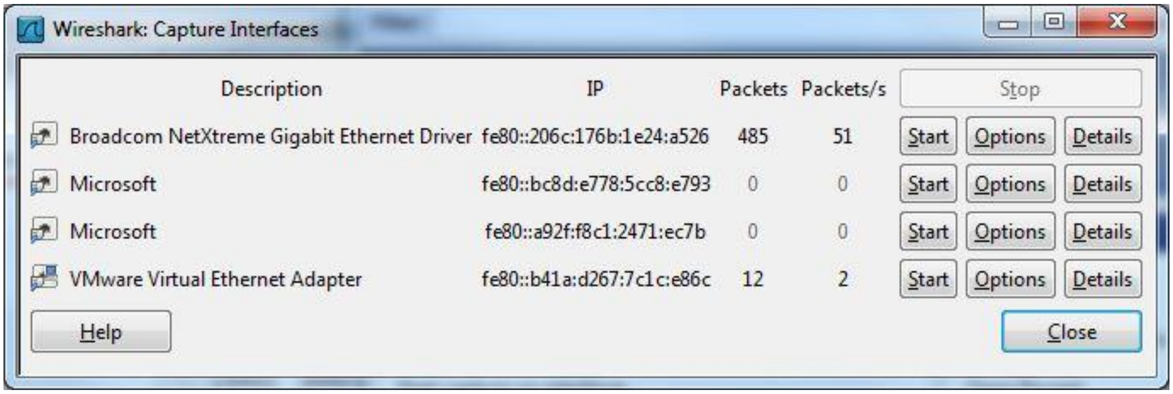
2. In **Capture Filter**, define whether specific traffic should be captured. For more information, click **Help**.

   Note: A capture filter can reduce the size of the capture file and can also be useful on high-traffic networks or for long-term capturing. The drawback to using capture filters is that other network traffic is not captured (which could be used to explain problems).

3.2 Starting the Capture

1. The capture can be started in several ways:

   a. In the startup window, click **Interface List**. Then, click the **Start** button for the interface that will be used.

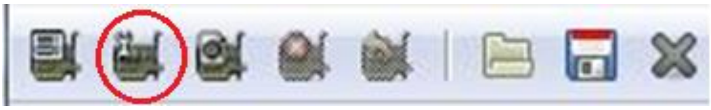


Wireshark: Capture Interfaces

Description	IP	Packets	Packets/s	Stop		
Broadcom NetXtreme Gigabit Ethernet Driver	fe80::206c:176b:1e24:a526	485	51	Start	Options	Details
Microsoft	fe80::bc8d:e778:5cc8:e793	0	0	Start	Options	Details
Microsoft	fe80::a92f:f8c1:2471:ec7b	0	0	Start	Options	Details
VMware Virtual Ethernet Adapter	fe80::b41a:d267:7c1c:e86c	12	2	Start	Options	Details

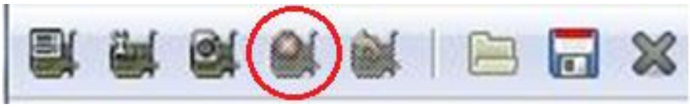
   b. In the main menu, click **Capture | Start**.

   c. Then, click the **Start a new live capture** toolbar icon.



   Note: Options b and c will start collecting from the default interface that was specified in Capture Options.

2. Once the test devices have communicated, stop the capture by clicking **Capture | Stop** in the main menu. Alternatively, click the **Stop the running live capture** toolbar icon.



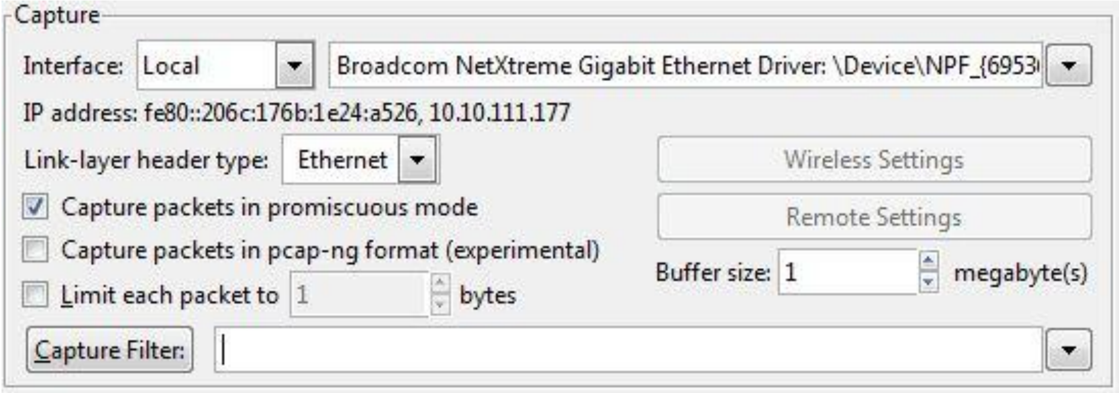
   Note: The packets should be displayed in the main window, and can be saved to disk.

3.3 Additional Capture Options

A packet capture should include everything because the TCP layer packets sometimes display the issue with the device instead of with the protocol-specific packets. Because captures can get very large on busy networks, users can specify additional capture options for managing the capture content.

3.3.1 Capture Filters

Capture Filters refine the capture, and may be entered by hand or through the Wireshark Filter Expression Builder.



Capture

Interface: Local — Broadcom NetXtreme Gigabit Ethernet Driver: \Device\NPF_{6953

IP address: fe80::206c:176b:1e24:a526, 10.10.111.177

Link-layer header type: Ethernet

☑ Capture packets in promiscuous mode

□ Capture packets in pcap-ng format (experimental)

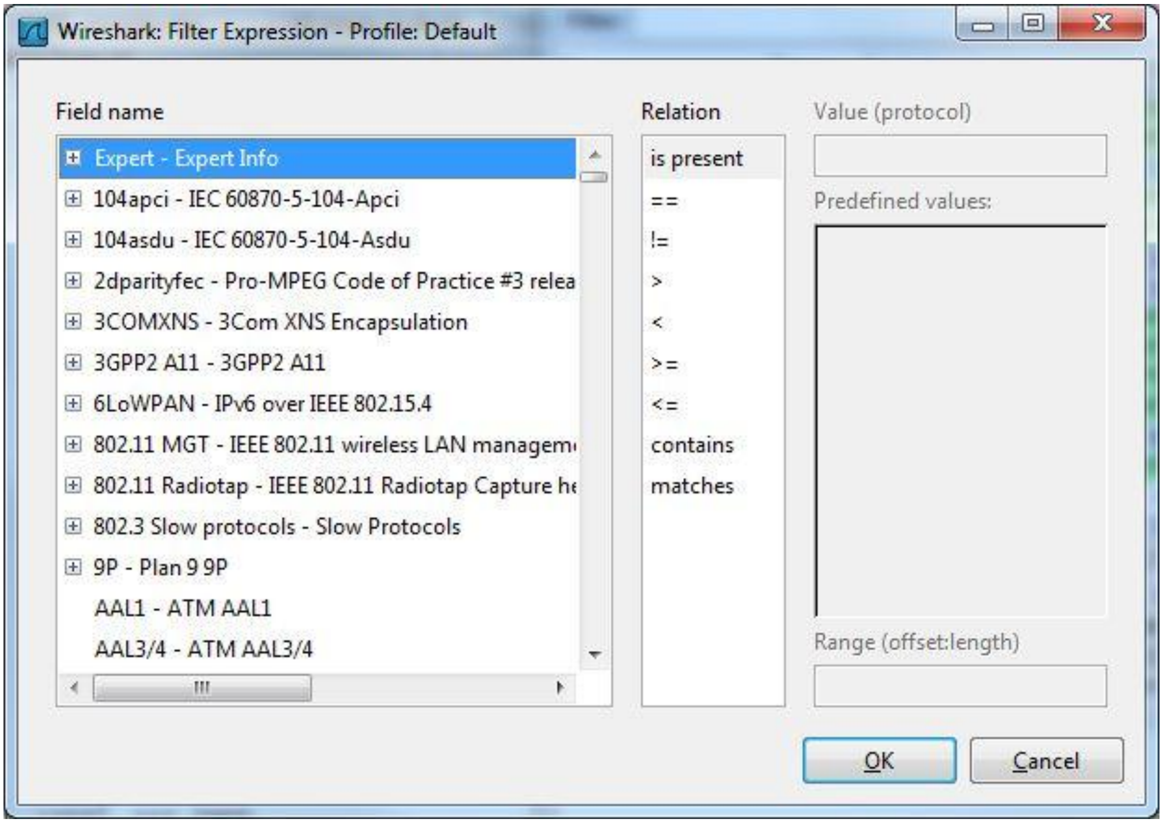
□ Limit each packet to 1 bytes

Buffer size: 1 megabyte(s)

Capture Filter:

A typical capture filter may be "ip.addr == 10.20.20.132". This filter would only perform a capture of those packets with a destination or source IP Address of "10.20.20.132"**.**

To use the Wireshark Filter Expression Builder, click **Capture Filter**.



Wireshark: Filter Expression - Profile: Default

Field name

- Expert - Expert Info
- 104apci - IEC 60870-5-104-Apci
- 104asdu - IEC 60870-5-104-Asdu
- 2dparityfec - Pro-MPEG Code of Practice #3 relea
- 3COMXNS - 3Com XNS Encapsulation
- 3GPP2 A11 - 3GPP2 A11
- 6LoWPAN - IPv6 over IEEE 802.15.4
- 802.11 MGT - IEEE 802.11 wireless LAN managem
- 802.11 Radiotap - IEEE 802.11 Radiotap Capture he
- 802.3 Slow protocols - Slow Protocols
- 9P - Plan 9 9P
- AAL1 - ATM AAL1
- AAL3/4 - ATM AAL3/4

Relation

- is present
- ==
- !=
- >
- <
- >=
- <=
- contains
- matches

Value (protocol)

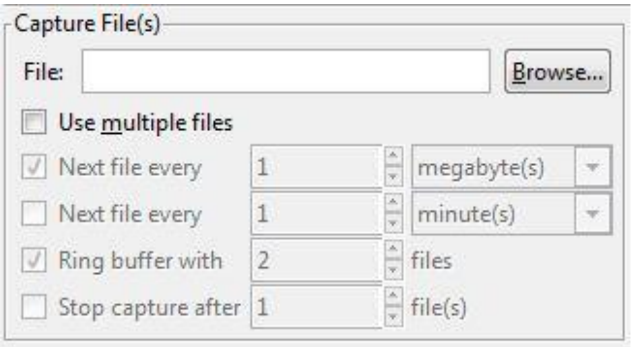
Predefined values:

Range (offset:length)

Note: For information on building filters with the Filter Expression Builder, refer to the Wireshark help documentation.

3.3.2 Capture Files

Users can specify the capture file name and location, and may also break it up into more manageable pieces. Creating files of a smaller size or duration makes it easier to send them as attachments.



Capture File(s)

File:

□ Use multiple files

☑ Next file every 1 megabyte(s)

□ Next file every 1 minute(s)

☑ Ring buffer with 2 files

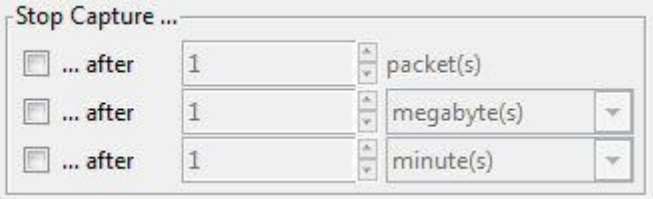
□ Stop capture after 1 file(s)

Descriptions of the parameters are as follows:

- **File:** This parameter specifies the capture's file name. It is blank by default. If left blank, the capture data will be stored in a temporary file.
- **Use Multiple Files:** When checked, Wireshark will automatically switch to a new file when a specific trigger condition is reached.
- **Next file every *n* megabyte(s):** This parameter will switch to the next file after the specified number of bytes, kilobytes, megabytes, or gigabytes have been captured. It is only available when Use Multiple Files is enabled. The default setting is 1 megabyte.
- **Next file every *n* minute(s):** This parameter will switch to the next file after the specified number of seconds, minutes, hours, or days have elapsed. It is only available when Use Multiple Files is enabled. The default setting is 1 minute.
- **Ring buffer with *n* files:** This parameter will form a ring buffer of the capture files using the specified number of files. It is only available when Use Multiple Files is enabled. The default setting is 2 files.
- **Stop capture after *n* file(s):** This parameter will stop capturing once it has switched to the next file the specified number of times. It is only available when Use Multiple Files is enabled. The default setting is 1 file.

3.3.3 Stop Capture

Users can also specify when to stop captures. When running an unmonitored capture, it is important to specify a limit as to not overload system memory.



Stop Capture ...

□ ... after 1 packet(s)

□ ... after 1 megabyte(s)

□ ... after 1 minute(s)

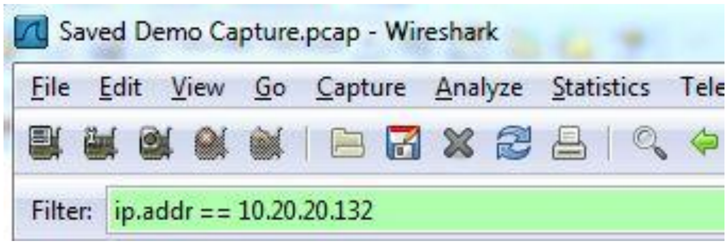
Descriptions of the parameters are as follows:

- **After *n* packet(s):** When checked, this option will stop capturing after the specified number of packets have been captured.

- **After *n* megabytes(s):** When checked, this option will stop capturing after the specified number of bytes, kilobytes, megabytes, or gigabytes have been captured. This option is disabled when Use Multiple Files is selected.
- **After *n* minute(s):** When checked, this option will stop capturing after the specified number of seconds, minutes, hours, or days have elapsed.

4. Verification and Analysis

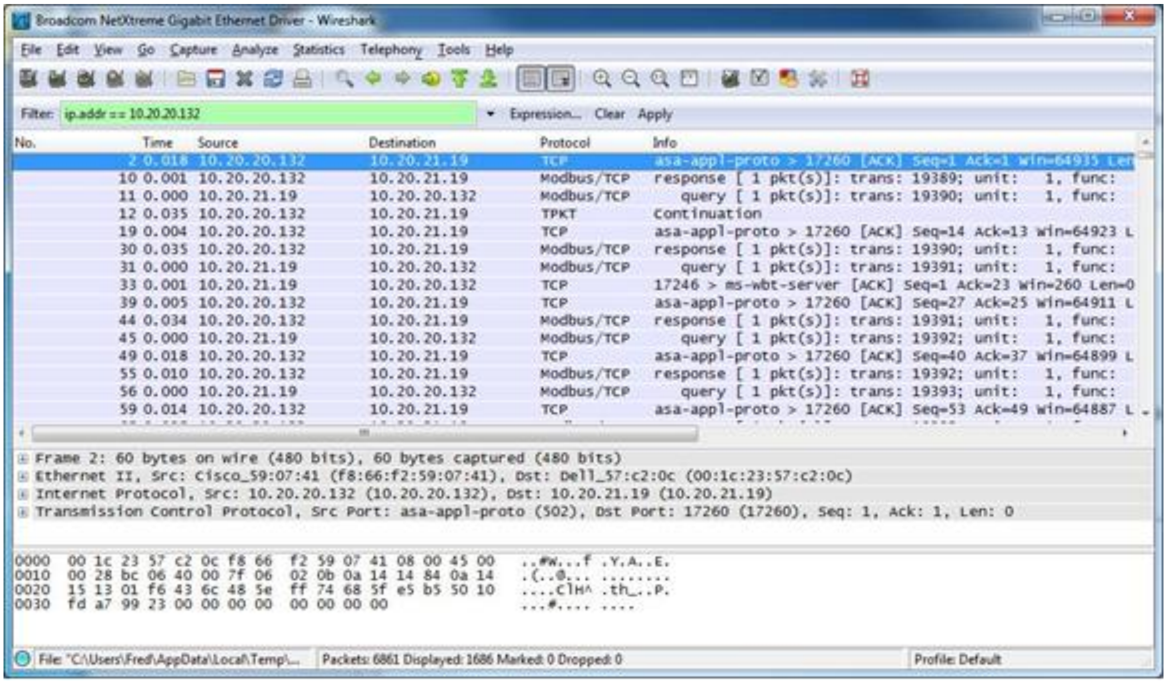
1. To start, examine the capture and verify that the communications were captured.
2. If extraneous packets from other devices or software were captured, use a display filter to locate the communications that were under test. For example, if a device's IP address is "10.20.20.132," enter "ip.addr == 10.20.20.132" into **Filter**. Then, click **Apply**.



Filter: ip.addr == 10.20.20.132

Note: If the display does not show any packets, there may be a problem with the capture setup.

3. Once the packets have been verified, users can start analysis.



Filter: ip.addr == 10.20.20.132

No.	Time	Source	Destination	Protocol	Info
2	0.018	10.20.20.132	10.20.21.19	TCP	asa-appl-proto > 17260 [ACK] Seq=1 Ack=1 Win=64935 Len
10	0.001	10.20.20.132	10.20.21.19	Modbus/TCP	response [1 pkt(s)]: trans: 19389; unit: 1, func:
11	0.000	10.20.21.19	10.20.20.132	Modbus/TCP	query [1 pkt(s)]: trans: 19390; unit: 1, func:
12	0.035	10.20.20.132	10.20.21.19	TPKT	Continuation
19	0.004	10.20.20.132	10.20.21.19	TCP	asa-appl-proto > 17260 [ACK] Seq=14 Ack=13 Win=64923 L
30	0.035	10.20.20.132	10.20.21.19	Modbus/TCP	response [1 pkt(s)]: trans: 19390; unit: 1, func:
31	0.000	10.20.21.19	10.20.20.132	Modbus/TCP	query [1 pkt(s)]: trans: 19391; unit: 1, func:
33	0.001	10.20.21.19	10.20.20.132	TCP	17246 > ms-wbt-server [ACK] Seq=1 Ack=23 Win=260 Len=0
39	0.005	10.20.20.132	10.20.21.19	TCP	asa-appl-proto > 17260 [ACK] Seq=27 Ack=25 Win=64911 L
44	0.034	10.20.20.132	10.20.21.19	Modbus/TCP	response [1 pkt(s)]: trans: 19391; unit: 1, func:
45	0.000	10.20.21.19	10.20.20.132	Modbus/TCP	query [1 pkt(s)]: trans: 19392; unit: 1, func:
49	0.018	10.20.20.132	10.20.21.19	TCP	asa-appl-proto > 17260 [ACK] Seq=40 Ack=37 Win=64899 L
55	0.010	10.20.20.132	10.20.21.19	Modbus/TCP	response [1 pkt(s)]: trans: 19392; unit: 1, func:
56	0.000	10.20.21.19	10.20.20.132	Modbus/TCP	query [1 pkt(s)]: trans: 19393; unit: 1, func:
59	0.014	10.20.20.132	10.20.21.19	TCP	asa-appl-proto > 17260 [ACK] Seq=53 Ack=49 Win=64887 L

```
⊞ Frame 2: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
⊞ Ethernet II, Src: Cisco_59:07:41 (f8:66:f2:59:07:41), Dst: Dell_57:c2:0c (00:1c:23:57:c2:0c)
⊞ Internet Protocol, Src: 10.20.20.132 (10.20.20.132), Dst: 10.20.21.19 (10.20.21.19)
⊞ Transmission Control Protocol, Src Port: asa-appl-proto (502), Dst Port: 17260 (17260), Seq: 1, Ack: 1, Len: 0
```

```
0000  00 1c 23 57 c2 0c f8 66  f2 59 07 41 08 00 45 00   ..#W...f .Y.A..E.
0010  00 28 bc 06 40 00 7f 06  02 0b 0a 14 14 84 0a 14   .(..@... ........
0020  15 13 01 f6 43 6c 48 5e  ff 74 68 5f e5 b5 50 10   ....ClH^ .th_..P.
0030  fd a7 99 23 00 00 00 00  00 00 00 00               ...#.... ....
```

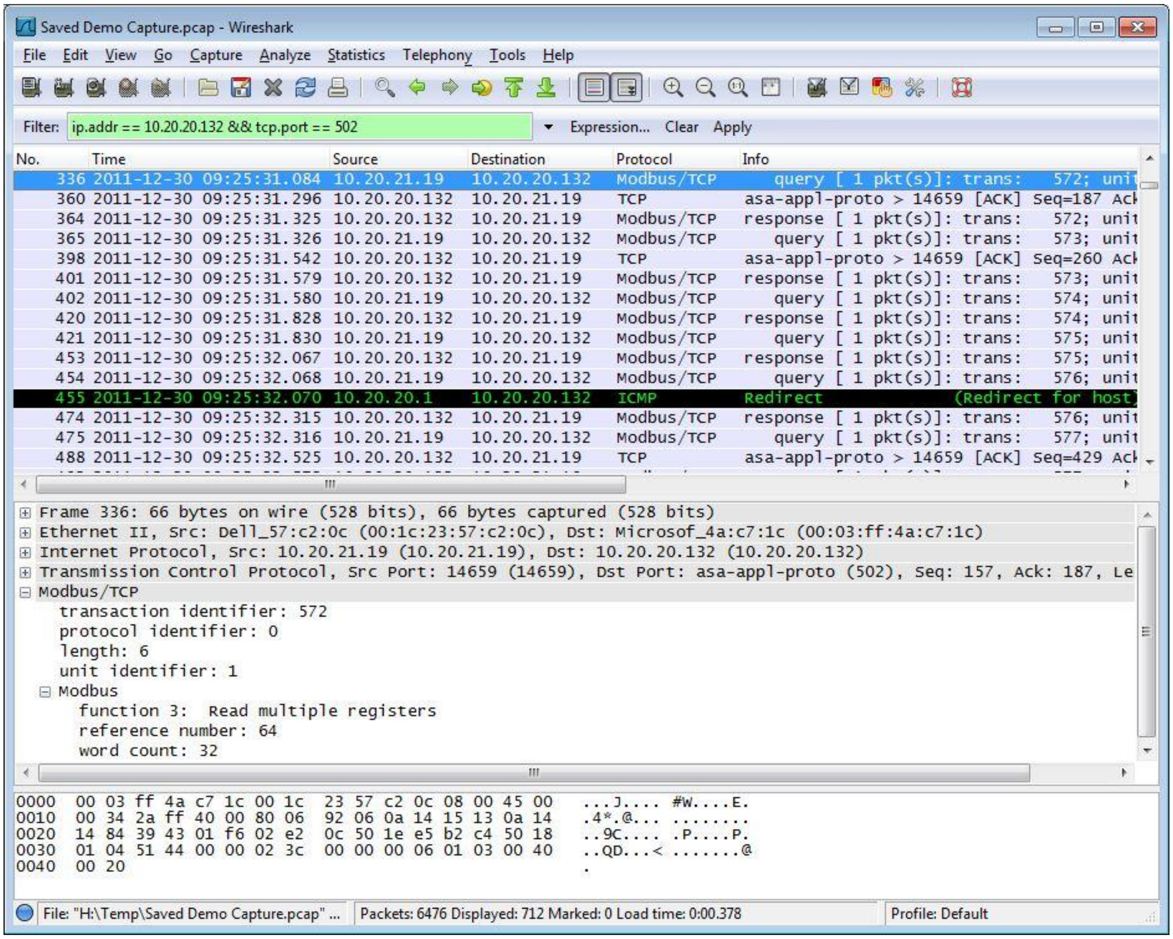
File: "C:\Users\Fred\AppData\Local\Temp\... Packets: 6861 Displayed: 1686 Marked: 0 Dropped: 0 Profile: Default

4.1 Additional Display Options

4.1.1 Compound Display Filters

One common display filter for analysis is ”ip.addr == 10.20.20.132 && tcp.port == 502”. This will only show traffic to or from TCP port 502 for the specified address.

Note: TCP port 502 is the standard port for Modbus/TCP Ethernet devices.


Saved Demo Capture.pcap - Wireshark
File Edit View Go Capture Analyze Statistics Telephony Tools Help
Expression... Clear Apply

Filter: ip.addr == 10.20.20.132 && tcp.port == 502

```
No.     Time                    Source          Destination     Protocol     Info
    336 2011-12-30 09:25:31.084 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:    572; unit
    360 2011-12-30 09:25:31.296 10.20.20.132    10.20.21.19     TCP          asa-appl-proto > 14659 [ACK] Seq=187 Ack
    364 2011-12-30 09:25:31.325 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:    572; unit
    365 2011-12-30 09:25:31.326 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:    573; unit
    398 2011-12-30 09:25:31.542 10.20.20.132    10.20.21.19     TCP          asa-appl-proto > 14659 [ACK] Seq=260 Ack
    401 2011-12-30 09:25:31.579 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:    573; unit
    402 2011-12-30 09:25:31.580 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:    574; unit
    420 2011-12-30 09:25:31.828 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:    574; unit
    421 2011-12-30 09:25:31.830 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:    575; unit
    453 2011-12-30 09:25:32.067 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:    575; unit
    454 2011-12-30 09:25:32.068 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:    576; unit
    455 2011-12-30 09:25:32.070 10.20.20.1      10.20.20.132    ICMP         Redirect                   (Redirect for host)
    474 2011-12-30 09:25:32.315 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:    576; unit
    475 2011-12-30 09:25:32.316 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:    577; unit
    488 2011-12-30 09:25:32.525 10.20.20.132    10.20.21.19     TCP          asa-appl-proto > 14659 [ACK] Seq=429 Ack
```

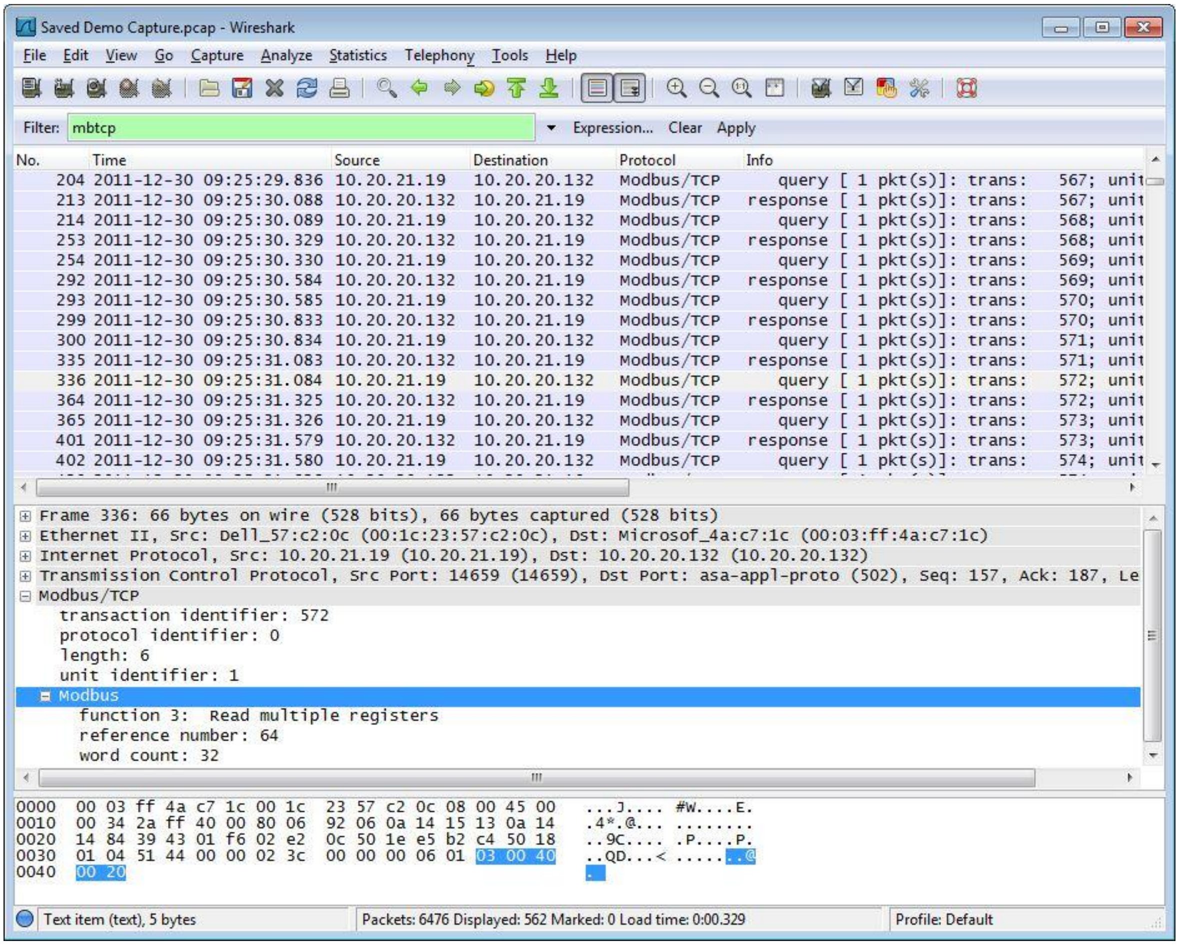
```
⊞ Frame 336: 66 bytes on wire (528 bits), 66 bytes captured (528 bits)
⊞ Ethernet II, Src: Dell_57:c2:0c (00:1c:23:57:c2:0c), Dst: Microsof_4a:c7:1c (00:03:ff:4a:c7:1c)
⊞ Internet Protocol, Src: 10.20.21.19 (10.20.21.19), Dst: 10.20.20.132 (10.20.20.132)
⊞ Transmission Control Protocol, Src Port: 14659 (14659), Dst Port: asa-appl-proto (502), Seq: 157, Ack: 187, Le
⊟ Modbus/TCP
    transaction identifier: 572
    protocol identifier: 0
    length: 6
    unit identifier: 1
  ⊟ Modbus
      function 3:  Read multiple registers
      reference number: 64
      word count: 32
```

```
0000  00 03 ff 4a c7 1c 00 1c  23 57 c2 0c 08 00 45 00   ...J.... #W....E.
0010  00 34 2a ff 40 00 80 06  92 06 0a 14 15 13 0a 14   .4*.@... ........
0020  14 84 39 43 01 f6 02 e2  0c 50 1e e5 b2 c4 50 18   ..9C.... .P....P.
0030  01 04 51 44 00 00 02 3c  00 00 00 06 01 03 00 40   ..QD...< .......@
0040  00 20                                              .
```

File: "H:\Temp\Saved Demo Capture.pcap" ... | Packets: 6476 Displayed: 712 Marked: 0 Load time: 0:00.378 | Profile: Default

4.1.2 Protocol Filter

In cases where the protocol used to communicate with a device is known by Wireshark, the protocol name can be entered into the filter. For example, entering "mbtcp" will filter the display for Modbus TCP packets only.

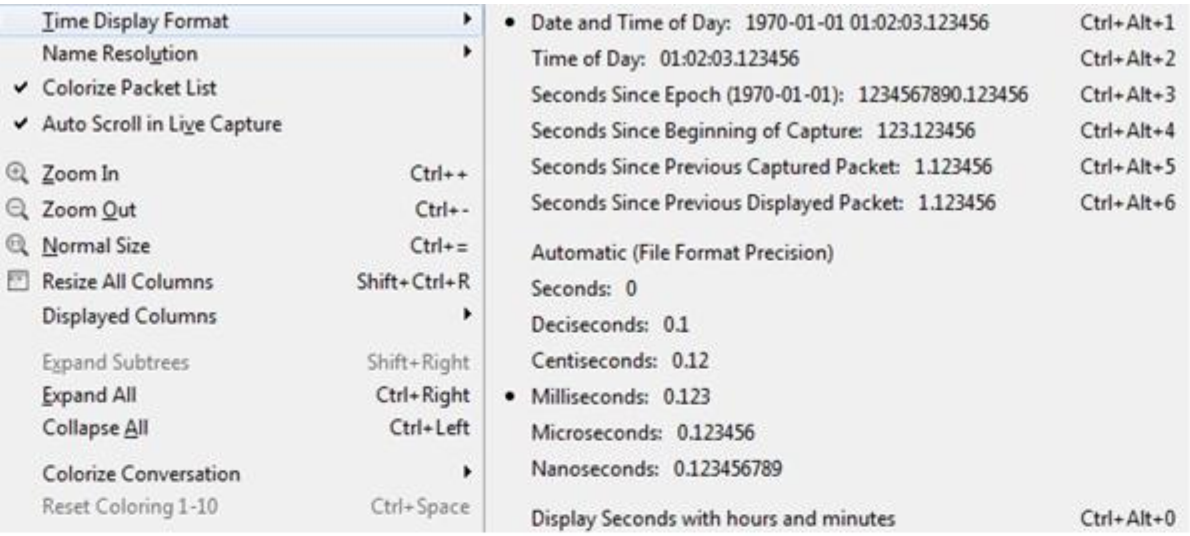

Saved Demo Capture.pcap - Wireshark
File Edit View Go Capture Analyze Statistics Telephony Tools Help
Expression... Clear Apply

```
Filter: mbtcp

No.  Time                     Source        Destination   Protocol    Info
 204 2011-12-30 09:25:29.836 10.20.21.19   10.20.20.132  Modbus/TCP     query [ 1 pkt(s)]: trans:   567; unit
 213 2011-12-30 09:25:30.088 10.20.20.132  10.20.21.19   Modbus/TCP  response [ 1 pkt(s)]: trans:   567; unit
 214 2011-12-30 09:25:30.089 10.20.21.19   10.20.20.132  Modbus/TCP     query [ 1 pkt(s)]: trans:   568; unit
 253 2011-12-30 09:25:30.329 10.20.20.132  10.20.21.19   Modbus/TCP  response [ 1 pkt(s)]: trans:   568; unit
 254 2011-12-30 09:25:30.330 10.20.21.19   10.20.20.132  Modbus/TCP     query [ 1 pkt(s)]: trans:   569; unit
 292 2011-12-30 09:25:30.584 10.20.20.132  10.20.21.19   Modbus/TCP  response [ 1 pkt(s)]: trans:   569; unit
 293 2011-12-30 09:25:30.585 10.20.21.19   10.20.20.132  Modbus/TCP     query [ 1 pkt(s)]: trans:   570; unit
 299 2011-12-30 09:25:30.833 10.20.20.132  10.20.21.19   Modbus/TCP  response [ 1 pkt(s)]: trans:   570; unit
 300 2011-12-30 09:25:30.834 10.20.21.19   10.20.20.132  Modbus/TCP     query [ 1 pkt(s)]: trans:   571; unit
 335 2011-12-30 09:25:31.083 10.20.20.132  10.20.21.19   Modbus/TCP  response [ 1 pkt(s)]: trans:   571; unit
 336 2011-12-30 09:25:31.084 10.20.21.19   10.20.20.132  Modbus/TCP     query [ 1 pkt(s)]: trans:   572; unit
 364 2011-12-30 09:25:31.325 10.20.20.132  10.20.21.19   Modbus/TCP  response [ 1 pkt(s)]: trans:   572; unit
 365 2011-12-30 09:25:31.326 10.20.21.19   10.20.20.132  Modbus/TCP     query [ 1 pkt(s)]: trans:   573; unit
 401 2011-12-30 09:25:31.579 10.20.20.132  10.20.21.19   Modbus/TCP  response [ 1 pkt(s)]: trans:   573; unit
 402 2011-12-30 09:25:31.580 10.20.21.19   10.20.20.132  Modbus/TCP     query [ 1 pkt(s)]: trans:   574; unit

⊞ Frame 336: 66 bytes on wire (528 bits), 66 bytes captured (528 bits)
⊞ Ethernet II, Src: Dell_57:c2:0c (00:1c:23:57:c2:0c), Dst: Microsof_4a:c7:1c (00:03:ff:4a:c7:1c)
⊞ Internet Protocol, Src: 10.20.21.19 (10.20.21.19), Dst: 10.20.20.132 (10.20.20.132)
⊞ Transmission Control Protocol, Src Port: 14659 (14659), Dst Port: asa-appl-proto (502), Seq: 157, Ack: 187, Le
⊟ Modbus/TCP
    transaction identifier: 572
    protocol identifier: 0
    length: 6
    unit identifier: 1
  ⊟ Modbus
      function 3:  Read multiple registers
      reference number: 64
      word count: 32

0000  00 03 ff 4a c7 1c 00 1c  23 57 c2 0c 08 00 45 00   ...J.... #W....E.
0010  00 34 2a ff 40 00 80 06  92 06 0a 14 15 13 0a 14   .4*.@... ........
0020  14 84 39 43 01 f6 02 e2  0c 50 1e e5 b2 c4 50 18   ..9C.... .P....P.
0030  01 04 51 44 00 00 02 3c  00 00 00 06 01 03 00 40   ..QD...< .......@
0040  00 20                                              .

Text item (text), 5 bytes | Packets: 6476 Displayed: 562 Marked: 0 Load time: 0:00.329 | Profile: Default
```

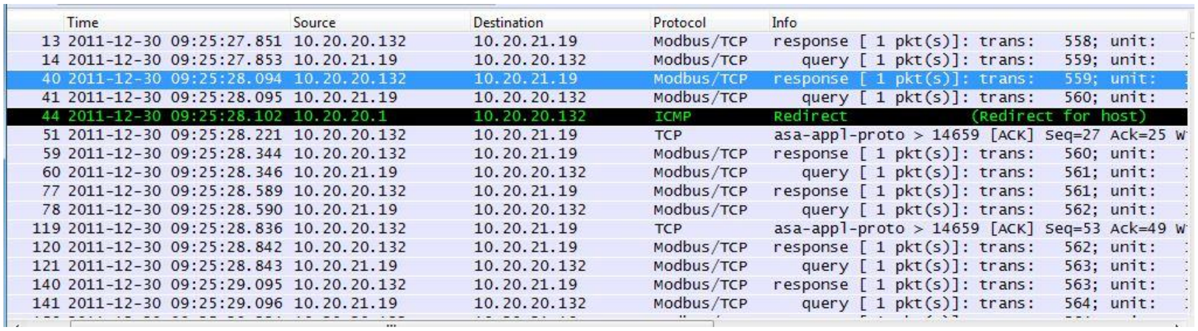
4.1.3 Time Formats

The Time Display Format can be used to view when packets were captured by Wireshark. The default format is **Date and Time of Day**.

Note: To change the default setting, click **View | Time Display Format**.


Time Display Format ▸
Name Resolution ▸
✔ Colorize Packet List
✔ Auto Scroll in Live Capture
Zoom In Ctrl++
Zoom Out Ctrl+-
Normal Size Ctrl+=
Resize All Columns Shift+Ctrl+R
Displayed Columns ▸
Expand Subtrees Shift+Right
Expand All Ctrl+Right
Collapse All Ctrl+Left
Colorize Conversation ▸
Reset Coloring 1-10 Ctrl+Space
● Date and Time of Day: 1970-01-01 01:02:03.123456 Ctrl+Alt+1
Time of Day: 01:02:03.123456 Ctrl+Alt+2
Seconds Since Epoch (1970-01-01): 1234567890.123456 Ctrl+Alt+3
Seconds Since Beginning of Capture: 123.123456 Ctrl+Alt+4
Seconds Since Previous Captured Packet: 1.123456 Ctrl+Alt+5
Seconds Since Previous Displayed Packet: 1.123456 Ctrl+Alt+6
Automatic (File Format Precision)
Seconds: 0
Deciseconds: 0.1
Centiseconds: 0.12
● Milliseconds: 0.123
Microseconds: 0.123456
Nanoseconds: 0.123456789
Display Seconds with hours and minutes Ctrl+Alt+0

The image below displays a Wireshark capture using Date and Time of Day Format.

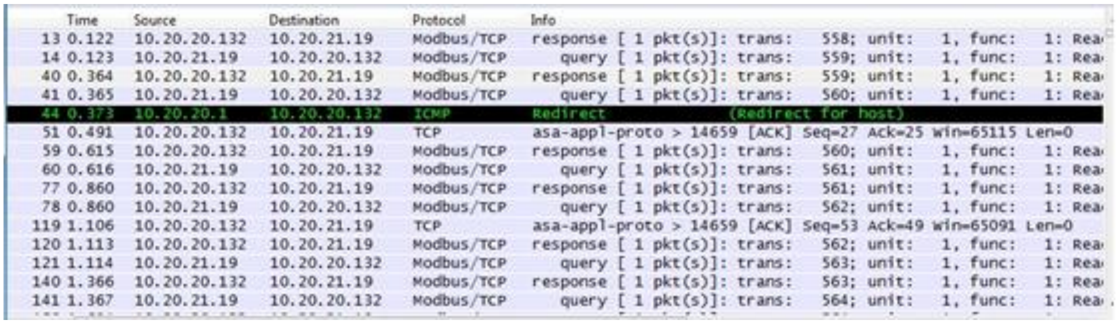


```
     Time                       Source          Destination     Protocol     Info
  13 2011-12-30 09:25:27.851 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:   558; unit:
  14 2011-12-30 09:25:27.853 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:   559; unit:
  40 2011-12-30 09:25:28.094 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:   559; unit:
  41 2011-12-30 09:25:28.095 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:   560; unit:
  44 2011-12-30 09:25:28.102 10.20.20.1      10.20.20.132    ICMP         Redirect            (Redirect for host)
  51 2011-12-30 09:25:28.221 10.20.20.132    10.20.21.19     TCP          asa-appl-proto > 14659 [ACK] Seq=27 Ack=25 W
  59 2011-12-30 09:25:28.344 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:   560; unit:
  60 2011-12-30 09:25:28.346 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:   561; unit:
  77 2011-12-30 09:25:28.589 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:   561; unit:
  78 2011-12-30 09:25:28.590 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:   562; unit:
 119 2011-12-30 09:25:28.836 10.20.20.132    10.20.21.19     TCP          asa-appl-proto > 14659 [ACK] Seq=53 Ack=49 W
 120 2011-12-30 09:25:28.842 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:   562; unit:
 121 2011-12-30 09:25:28.843 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:   563; unit:
 140 2011-12-30 09:25:29.095 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:   563; unit:
 141 2011-12-30 09:25:29.096 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:   564; unit:
```

Note: The Wireshark capture is often used to investigate possible issues with the timing of communications. In those situations, users can change the display format. Options are as follows:

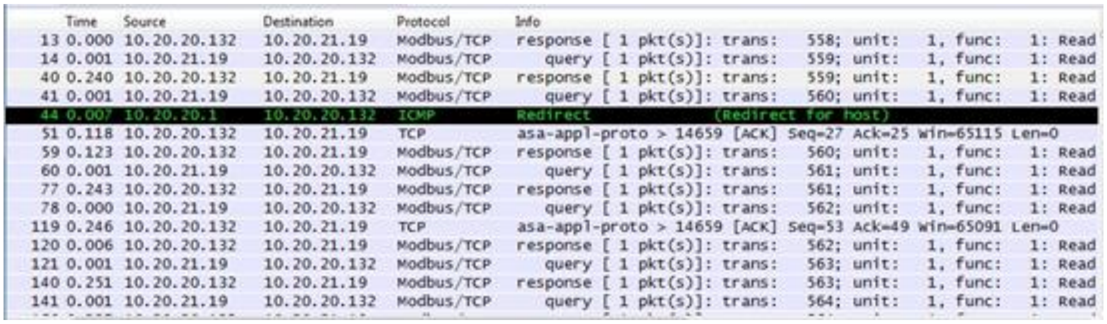
- **Seconds Since Beginning of Capture:** This option shows the seconds and milliseconds of each packet since the capture started.
- **Seconds Since Previous Displayed Packet:** This option shows the time in seconds and milliseconds since the previous packet. This is very useful when used with a display filter to isolate communications to a single device.

4.1.3.1 Seconds Since Beginning of Capture



```
     Time  Source          Destination     Protocol     Info
  13 0.122 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:   558; unit:   1, func:   1: Rea
  14 0.123 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:   559; unit:   1, func:   1: Rea
  40 0.364 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:   559; unit:   1, func:   1: Rea
  41 0.365 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:   560; unit:   1, func:   1: Rea
  44 0.373 10.20.20.1      10.20.20.132    ICMP         Redirect            (Redirect for host)
  51 0.491 10.20.20.132    10.20.21.19     TCP          asa-appl-proto > 14659 [ACK] Seq=27 Ack=25 Win=65115 Len=0
  59 0.615 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:   560; unit:   1, func:   1: Rea
  60 0.616 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:   561; unit:   1, func:   1: Rea
  77 0.860 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:   561; unit:   1, func:   1: Rea
  78 0.860 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:   562; unit:   1, func:   1: Rea
 119 1.106 10.20.20.132    10.20.21.19     TCP          asa-appl-proto > 14659 [ACK] Seq=53 Ack=49 Win=65091 Len=0
 120 1.113 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:   562; unit:   1, func:   1: Rea
 121 1.114 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:   563; unit:   1, func:   1: Rea
 140 1.366 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:   563; unit:   1, func:   1: Rea
 141 1.367 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:   564; unit:   1, func:   1: Rea
```

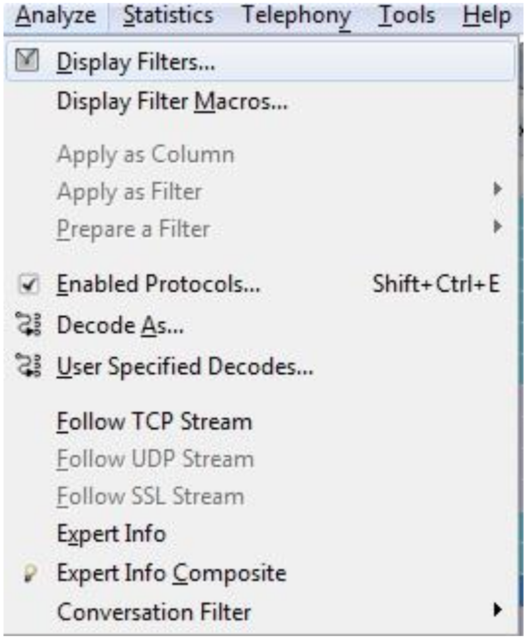
4.1.3.1 Seconds from Previous Displayed Packet



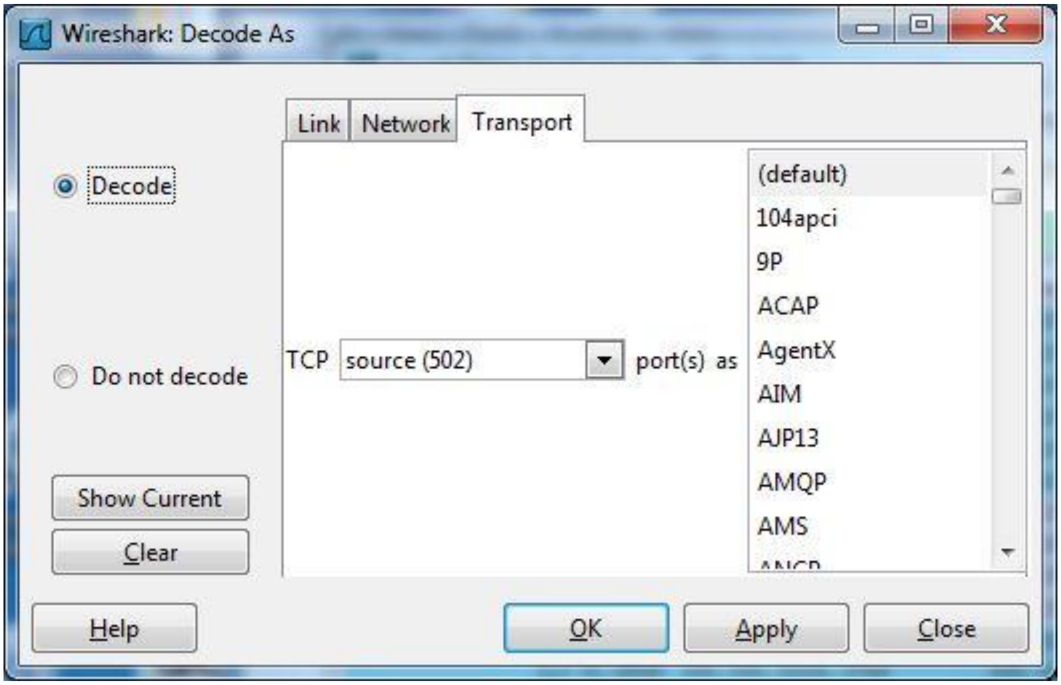
```
     Time  Source          Destination     Protocol     Info
  13 0.000 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:   558; unit:   1, func:   1: Read
  14 0.001 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:   559; unit:   1, func:   1: Read
  40 0.240 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:   559; unit:   1, func:   1: Read
  41 0.001 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:   560; unit:   1, func:   1: Read
  44 0.007 10.20.20.1      10.20.20.132    ICMP         Redirect            (Redirect for host)
  51 0.118 10.20.20.132    10.20.21.19     TCP          asa-appl-proto > 14659 [ACK] Seq=27 Ack=25 Win=65115 Len=0
  59 0.123 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:   560; unit:   1, func:   1: Read
  60 0.001 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:   561; unit:   1, func:   1: Read
  77 0.243 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:   561; unit:   1, func:   1: Read
  78 0.000 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:   562; unit:   1, func:   1: Read
 119 0.246 10.20.20.132    10.20.21.19     TCP          asa-appl-proto > 14659 [ACK] Seq=53 Ack=49 Win=65091 Len=0
 120 0.006 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:   562; unit:   1, func:   1: Read
 121 0.001 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:   563; unit:   1, func:   1: Read
 140 0.251 10.20.20.132    10.20.21.19     Modbus/TCP   response [ 1 pkt(s)]: trans:   563; unit:   1, func:   1: Read
 141 0.001 10.20.21.19     10.20.20.132    Modbus/TCP      query [ 1 pkt(s)]: trans:   564; unit:   1, func:   1: Read
```

4.2 Using the Decode As Wizard

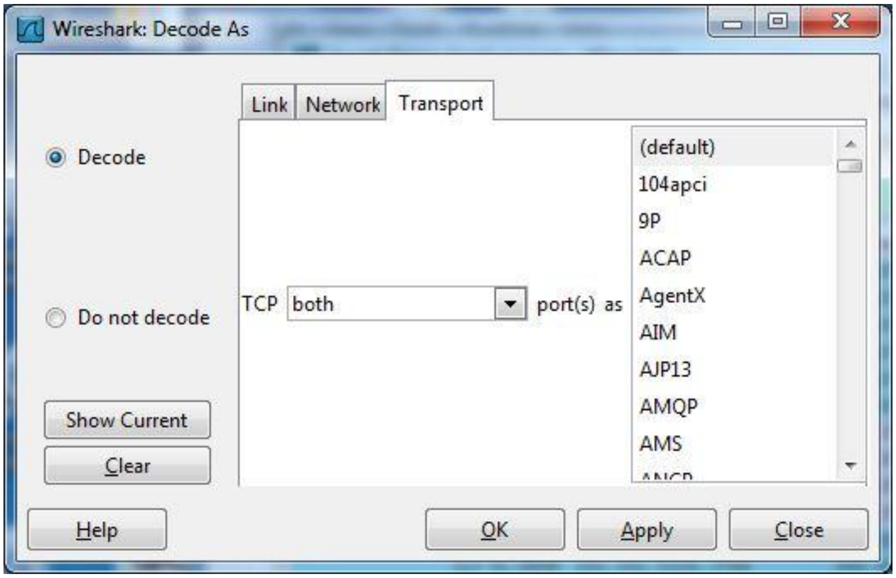
Wireshark provides developers with a method of decoding packets. To access it, click **Analyze** and then select **Decode As**.



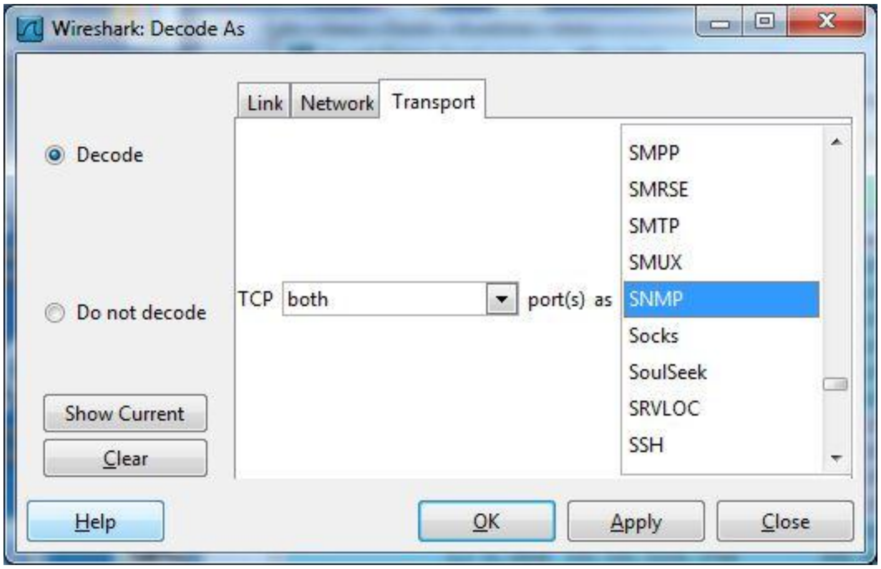
This wizard lets users specify how to decode packets (if Wireshark has not done so already).



For example, changing the Transport setting gets both source and destination ports.



In this example, the SNMP protocol was selected.



To view or clear the decodes, click **Analyze | User Specified Decodes**.

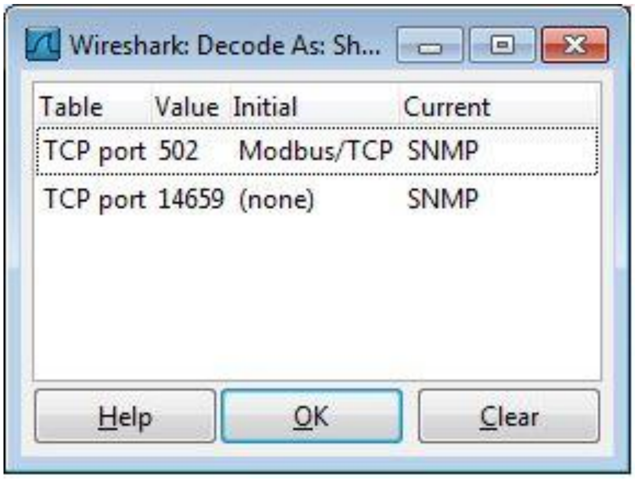


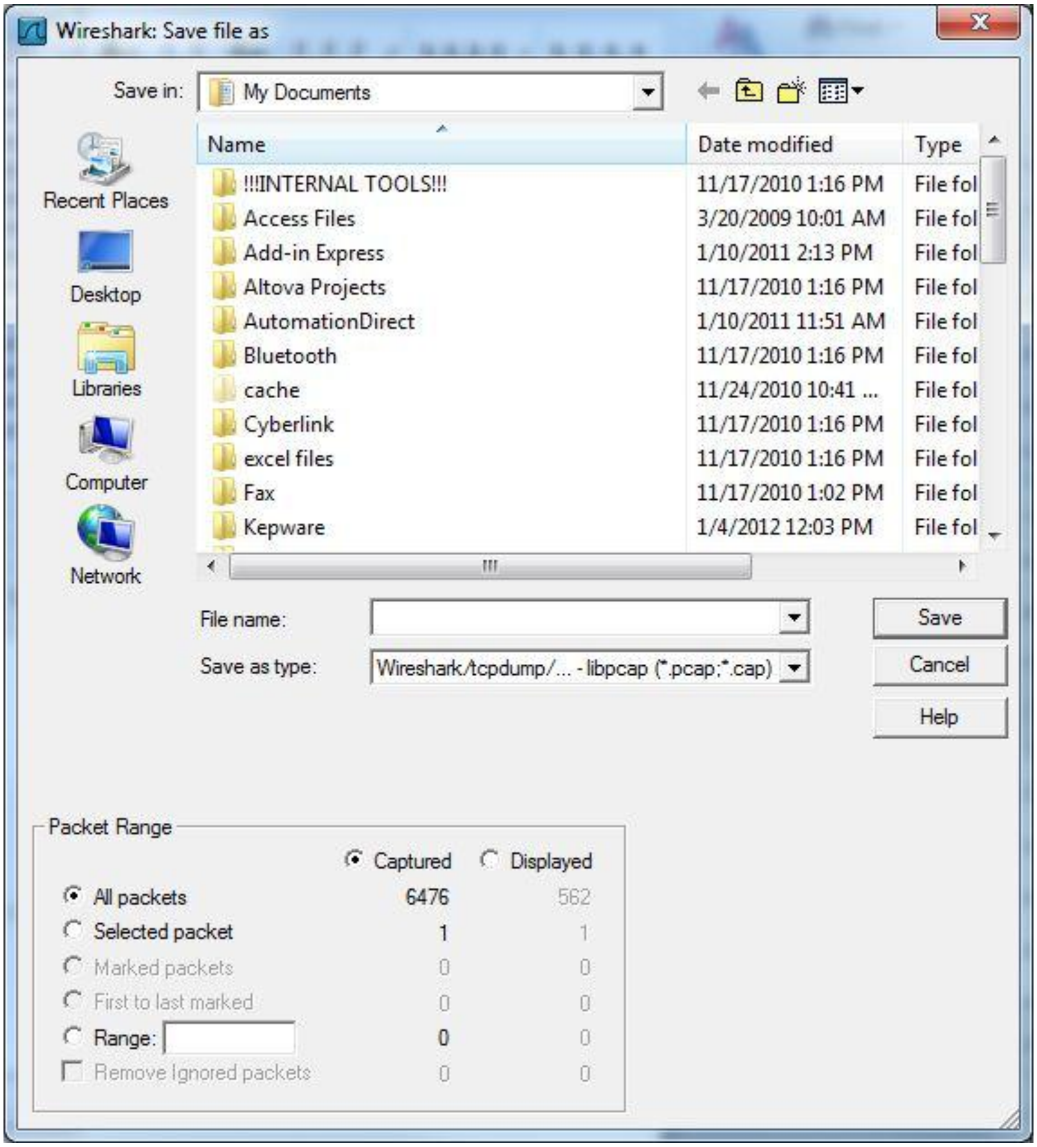
Table	Value	Initial	Current
TCP port	502	Modbus/TCP	SNMP
TCP port	14659	(none)	SNMP

5. Saving the Capture

Once the capture has been completed, users have several options in saving it. Kepware normally prefers to receive the raw capture file; however, there may be security issues that require the file to be filtered to prevent confidential information from being sent. For more information, refer to the instructions below.

Note: Users that utilized the Capture Multiple Files option can send the captured files to the Technical Support Case Contact. Users that utilized a single file (or who filtered a previous capture with a filter) should follow the instructions below.

1. To start, click **File | Save** or **File | Save As**.



Save in: My Documents

Name	Date modified	Type
!!!INTERNAL TOOLS!!!	11/17/2010 1:16 PM	File fol
Access Files	3/20/2009 10:01 AM	File fol
Add-in Express	1/10/2011 2:13 PM	File fol
Altova Projects	11/17/2010 1:16 PM	File fol
AutomationDirect	1/10/2011 11:51 AM	File fol
Bluetooth	11/17/2010 1:16 PM	File fol
cache	11/24/2010 10:41 ...	File fol
Cyberlink	11/17/2010 1:16 PM	File fol
excel files	11/17/2010 1:16 PM	File fol
Fax	11/17/2010 1:02 PM	File fol
Kepware	1/4/2012 12:03 PM	File fol

File name:

Save as type: Wireshark/tcpdump/... - libpcap (*.pcap;*.cap)

Packet Range

	● Captured	○ Displayed
● All packets	6476	562
○ Selected packet	1	1
○ Marked packets	0	0
○ First to last marked	0	0
○ Range:	0	0
□ Remove Ignored packets	0	0

2. Select **Captured** to save all the captured packets.
3. Select **Displayed** to save the packets displayed with the Display Filter.
4. Then, specify the other options as desired and click **Save**.

6. Summary

At this point, users should have a better understanding of using a software utility like Wireshark to capture packets being sent or received by the PC's Ethernet card.