# **Wireshark Basics**

by Felix Kolwa

# **Prerequisites**

This lesson assumes basic knowledge of networking concepts.

### Introduction

This article will be covering Wireshark including the following topics:

- Getting Wireshark
- Software overview
- Basic filtering
- Example usage

# What is Wireshark

Wireshark is a network packet analyzer. It is used to capture data from a network and display its content. Being an analyzer, Wireshark can only be used to measure data but not manipulate or send it. Wireshark is open source and free which makes it one of the most popular network analyzer available.

# **Getting Wireshark**

Wireshark is available for Linux, Windows and Mac through the official website. Formore information about building Wireshark from source please take a look at the official developers guide.

# **Running Wireshark**

Depending on your operating system and user settings you might have to run Wireshark with admin privileges to capture packets on your network. If your welcome screen is blank and does not show any network interfaces it usually means that your user account is lacking the necessary access rights.

# **Solution** Cybus Learn > Build and improve

# Your first capture

File Edit View Go Capture Ana	lyze <u>S</u> tatistics Telephon <u>y W</u> ireless <u>T</u> ools <u>H</u> elp
Apply a display filter <ctrl-></ctrl->	
_	
W	elcome to Wireshark
Ca	pture
	sing this filter: 📙 Enter a capture filter
	wlp2s0
	any stranger in a second
	Loopback: lo
	virbr0
	br-7442a3d071fbbr-0d96d4235dfd
	docker0
	br-9b3be7ba9c6e
	br-a26398e576c9 br-3da59f2fb028
	br-5c29e0dd3a45
	br-134460644a64
	nflog
	nfqueue
	usbmon1usbmon2
	usbmon3
	usbmon4
00	Cisco remote capture: cisco
•	Random packet generator: randpkt
ŏ	UDP Listener remote capture: udpdump

Once Wireshark is started you will be greeted by a welcome screen like the one shown above listing all available network connections. A small traffic preview is shown next to the interface names so it is easy to distinguish between interfaces with or without direct network access. To finally start capturing data on your network you first have to select one or more of these network interfaces by simply clicking on them. To select multiple interfaces at once just hold down ctrl and select all interfaces you want to listen on. Once selected you can start recording packets by clicking the start icon in the top left of the user interface.

*wlp2s	0 (host	: www.goo	gle.o	com)												
<u>F</u> ile	<u>E</u> dit	View	<u>G</u> o	<u>C</u> apture	<u>A</u> nalyze	<u>S</u> tatis	stics	Telep	ohon <u>y</u>	<u>W</u> ire	eless ]	ools <u>H</u>	elp			
		6		0101 0110 0110	8	9	¢	•	È	1			Ð,	Q	⊜	
A	pply a	display f	ilter	<ctrl-></ctrl->	>											
No.	Tin	ne	. :	Source				Destina	ation		Proto	col	Length		Info	
<b></b> •	20.	0000000 0558725 0014848	76	192.168.2 216.58.2 192.168.2	14.100			2 <mark>16.58</mark> 192.16 216.58	8.2.1	56	ICMP ICMP ICMP			98	Echo	(ping) (ping) (ping)

The window will change to the main capturing view and immediately display everything passing the network on your selected capturing device as see below.

	Apply a display filter				
lo.	Time	Source	Destination	Protocol	Length Info
	728 6.134282722		13.32.14.247	TCP	66 35374 → 443
	729 6.136240091		192.168.2.156	TCP	1506 443 → 35374
	730 6.136310017		13.32.14.247	TCP	66 35374 → 443
	731 6.136485158		192.168.2.156	TLSv1.2	426 Application
	732 6.136509901	192.168.2.156	13.32.14.247	TCP	66 35374 → 443
	733 6.136552412 734 6.136580620	13.32.14.247	192.168.2.156	TLSv1.2 TCP	104 Application
	735 6.244507701		13.32.14.247	NBNS	66 35374 → 443 92 Name query
		fe80::60ff:4dff:fe4f:8c9c		SSDP	395 NOTIFY * HT
	737 6.653050614		224.0.0.251	MDNS	87 Standard qu
	738 7.570632088	192.0.73.2	192.168.2.156	TLSv1.2	93 Application
		192.168.2.156	192.0.73.2	TCP	54 40030 → 443
	740 7.571181576		192.0.73.2	TCP	54 40030 → 443
	741 7.571346782		192.168.2.156	TLSv1.2	78 Application
	742 7.571397672		192.0.73.2	TCP	54 40030 - 443
	743 7.571542058		192.168.2.156	TCP	60 443 → 40030
	744 7.571569381		192.0.73.2	тср	54 40030 → 443
	745 7.602044949		192.168.2.156	TCP	60 443 → 40030
	746 7.602134106		192.0.73.2	TCP	54 40030 → 443
		Siemens- 8e:55:f6	PN-MC 00:00:00	PN-DCP	60 Ident Req,
	748 7.781532158		192.168.2.255	BROWSER	216 Get Backup
	749 8.598646128	192.168.2.152	224.0.0.251	MDNS	87 Standard qu
	750 8.667917374	13.32.14.235	192.168.2.156	TLSv1.2	112 Application
	751 8.668391197	192.168.2.156	13.32.14.235	TCP	66 57746 → 443
	752 8.669157242	13.32.14.235	192.168.2.156	TLSv1.2	97 Encrypted A
	753 8.669230673	192.168.2.156	13.32.14.235	тср	54 57746 → 443
		13.32.14.235	192.168.2.156	TCP	66 443 → 57746
	755 8.669357184		13.32.14.235	TCP	54 57746 → 443
		13.32.14.235	192.168.2.156	TCP	66 443 → 57746
	757 8.691931242	192.168.2.156	13.32.14.235	тср	54 57746 → 443

Simple Service Discovery Protocol

Stop the current capturing process by clicking on the red stop button.

# **Filtering Traffic**

Even the smallest network will produce a lot of static data that can result in very large capture files. To avoid slowdowns you should not capture unfiltered network traffic. To do so open the capture configuration window by clicking on the cogwheel icon.

*wlp2s	0 (hos	t www.goo	ogle.	com)									
<u>F</u> ile	<u>E</u> dit	<u>V</u> iew	<u>G</u> o	<u>C</u> apture	<u>A</u> nalyze	<u>S</u> tatistic	s Telephony	<u>W</u> irele	ss <u>T</u> ools	<u>H</u> elp			
		6		0100 0100 0111	8	٩ «	Þ 🔿 警			Ð,	Q	₽,	
A	pply a	display f	filter	<ctrl- :<="" th=""><th>&gt;</th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th></ctrl->	>								
No.	Tir	me		Source			Destination		Protocol	Length	1	nfo	
-		0000000		192.168.			216.58.214.:		ICMP				(ping)
		0558725		216.58.2			192.168.2.1		ICMP			Echo	(ping)
	0.4	0014848		192.168.			216.58.214.		ICMP			Echo	(ping)

This will open the capture configuration menu. This menu provides options similar to those you already saw on the welcome screen. You can select network devices, set capture filters and configure the capturing process. This time we want to apply a filter before we start capturing data. Select the network interface of your choice and just type ,tcp' into the capture filter dialog box on the bottom of the configuration window like below.

nterface	Iraffic	Link-layer Header	Promis	s Snaplen	(I Buf
enpls0		Ethernet	✓	default	2
wlp2s0		Ethernet	V	default	2
any	1000	Linux cooked	V	default	2
Loopback: lo	17. <u></u>	Ethernet	~	default	2
▶ virbr0		Ethernet	~	default	2
docker0		Ethernet	~	default	2
br 89ecc31668c5		Ethernet	~	default	2
br-8c7d89a62ea7	·	Ethernet	1	default	2
br-1d5394d6272a		Ethernet	1	default	2
br-10c73b45721c		Ethernet	~	default	2
br-62a3a3644042		Ethernet	~	default	2
br-a8cf6745782f		Ethernet	~	default	2
bluetooth0		Bluetooth HCI UART transport layer plus pseudo-header	1	default	2
nflog		Linux netfilter log messages	~	default	2
nfqueue	Resource and the	Raw IPv4	~	default	2
usbmon1		DLT -1	V	default	2
uchmon?		DIT 1	.7	dotault	2
I Enable promiscuous mode	on all interfaces		Mar	nage Inter	aces.

Now when you now start capturing again only packets applicable to the tcp protocol filter are captured and displayed.

# **More on Filters**

Wireshark provides a powerful filter language which not only allows you to narrow down the packets you want to capture but also to sort, follow or even compare their content. This section will only scratch the surface of what is possible with Wireshark so for the time being please consult the Wireshark Wiki for further information about creating filters.

It is a common mistake to believe that capture filters and display filters work the same way in Wireshark. While capture filters change the outcome of the capturing process, display filters can be applied to already running capturing processes to narrow down what to display. Furthermore they use different filter language syntax.

#### Capture filter example

To narrow down our captured data to only include packets from a certain ip range: src net 192.168.2.0/24

# **Solution** Cybus Learn > Build and improve

#### **Display filter example**

The same can be done to filter the already captured data in the main window: ip.addr == 192.168.2.0/24

#### **Combining filters**

To find exactly what you are looking for on your network you can concatenate different filters. If you want to capture packets from a certain host and port you can simply add both filters together: host 192.168.2.100 and port 20

You can specify data that you want to explicitly exclude: host www.google.com and not (port 20 or port 80)

This would only capture data from a certain host which is not transferred on port 20 or 80.

### **Collecting data**

A standard example to see actual network traffic is to ping a host and collect the data.

Just run a capture and set the capture filter to the host you are going to ping (www.google.com would be a popular choice).

host www.google.com

Go ahead and start the capturing process. Without any connections to your host open the main window should stay empty for now.

Next open a terminal window and ping the host you specified in the capture filter. Within a few moments you should see the first packets.

Once you have captured some packets press the stop button.

# Analyzing at the data

After collecting data the user interface contains three main parts. Those being the packet list pane, the packet details pane and packet bytes pane.

On top is the packet list pane. This view displays a summary of all the captured packets. You can choose any of the packets by just selecting and the other two views will adapt to the selection. Go ahead and select any of the packets and notice how the other two views change.

# **Solution** Cybus Learn > Build and improve

The one in the middle is the packet details pane. It shows more details about the packets you select in the packet list pane.

Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0
Ethernet II, Src: LiteonTe\_35:1e:7f (cc:b0:da:35:1e:7f), Dst: Avm\_ca:1f:c4 (34:81:c4:ca:1f:c4)
Internet Protocol Version 4, Src: 192.168.2.156, Dst: 216.58.214.100
Internet Control Message Protocol

On the bottom the packet bytes pane displays the actual data transferred in the packets.

# **Cybus Learn** > Build and improve

0000	34	81	c4	са	1f	c4	СС	b0	da	35	1e	7f	08	00	45	00	4E.
0010	00	54	8d	63	40	00	40	01	3b	62	c0	a8	02	9c	d8	3a	.T.c@.@. ;b:
0020	d6	64	08	00	28	9d	4e	69	00	01	3a	са	7e	5b	00	00	.d(.Ni:.~[
0030	00	00	03	00	06	00	00	00	00	00	10	11	12	13	14	15	
0040	16	17	18	19	1a	1b	1c	1d	1e	1f	20	21	22	23	24	25	!"#\$%
0050	26	27	28	29	2a	2b	2c	2d	2e	2f	30	31	32	33	34	35	&'()*+,/012345
0060	36	37															67

Using these sections you can view the traffic and break it down for analysis.

### Summary

Wireshark is a powerful network packet analyzer. It offers everything you need to capture, filter and view your local network traffic. After reading through this article you should have all the basic knowledge necessary to create and filter simple captures.

#### LEARN MORE

Official Wireshark learn platform

Cybus is a specialist for secure IIoT Edge software, headquartered in Germany. Cybus Connectware serves smart factories as a universal Edge and DevOps hub. Machine builders and providers of IIoT services use the Cybus Connectware as a software-based gateway. As early as 2017, Cybus published the first secure industrial connector for machine data according to today's DIN SPEC 27070 standard. Industry analyst Gartner named Cybus a worldwide "Cool Vendor". Today, the company counts medium-sized and large companies from numerous industrial sectors such as mechanical engineering, automotive and aviation among its customers.

Cybus GmbH · Osterstraße 124 · 20255 Hamburg · Germany · www.cybus.io · hello@cybus.io · (+49) 40 228 58 68 51