

Packet Capture Using Wireshark



Made possible through support from the National Science Foundation (NSF) award number <u>1800929</u>

Objectives

- Discuss the purpose of packet capture software such as Wireshark.
- Use Wireshark to capture network data.
- Explain the different ways Wireshark can present and format captured data.
- Control the display and capture of network data using filters.
- Discuss various ways networks and network devices can be manipulated to allow the capture of network traffic.

Wireshark Overview

- What is Wireshark?
 - Wireshark is software that allows us to view all data being transmitted on a network
 - Wireshark allows us to view fully decoded data or view data in its raw (binary) format
 - Wireshark is free, open-source software
 - Wireshark is available for multiple platforms (Linux, MAC, Windows)
 - https://www.wireshark.org



Wireshark Simple Usage

- Download and install
- Needs to be run as the super-user or permissions need to be configured to allow regular user access



Wireshark Simple Usage

Select the interface to be used to capture the data

Welcome to Wireshark			
Capture			
using this filter: 📙 Enter a capture filter			 All interfaces shown *
cell-area-zone			
any bluetooth-monitor			
nflog nfqueue manufact-zone			
 Gisco remote capture: ciscodump DisplayPort AUX channel monitor capture: dpau 	 		
 Random packet generator: randpkt systemd Journal Export: sdjournal 	_		
 SSH remote capture: sshdump UDP Listener remote capture: udpdump 	-		

Wireshark Simple Usage

- Click the Start button to begin capture
- Click the Stop button to end capture





Wireshark Screen Layout

- When viewing a packet capture, the Wireshark screen is divided into three sections
- The top pane (packet list) shows an ordered list containing a summary of each packet captured

File Edit View Go Capture Analyze Sta	atistics Telephon <u>y W</u> ireless <u>T</u>	<u> </u> ools <u>H</u> elp
	९ 🗢 🔿 🖀 🚡 📃	
Apply a display filter <ctrl-></ctrl->		
No. Time Source	Destination Proto	ocol Lengti Info
<pre>4 1.001550699</pre>	255.255.255.255.055 18.0.105.202 DHCC Broadcast ARP Broadcast BR Broadcast ARP Broadcast BR Broadcast BR BR BR BR BR BR BR BR BR BR	342 DHCP Kequest - Iransaction ID 0x/affa64a 342 DHCP ACK - Transaction ID 0x/affa64a 42 WhC has 10, 0.255, 1022 Tell 10, 0.105, 202 42 WhC has 10, 0.255, 1027 Tel
Hardware type: Ethernet (1) Protocol type: IPv4 (0x0800) Hardware size: 6 Protocol size: 4 Opcode: request (1) Sender IMA address: VMware 41:0c:36 Sender IP address: 10.0.105.202 Target MAC address: 00:00:00_00:00: Target IP address: 10.0.255.102 0000 ff ff ff ff ff ff 00 0c 29 41 0c 3 0010 08 00 06 04 00 10 00 cc 29 41 0c 3 0020 00 00 00 00 00 00 00 a0 01 ff 66	6 (00:0c:29:41:0c:36) 00 (00:00:00:00:00:00) 16 08 06 00 01	j 1911

Wireshark Screen Layout

- When viewing a packet capture, the Wireshark screen is divided into three sections
- The middle pane (packet details) shows detailed and decoded data associated with the packet selected in the packet list pane
 - Some summary data listed in the packet details pane can be expanded to provide more detailed information about the section of the packet being displayed

<u>File Edit View Go</u> Capture A	analyze <u>S</u> tatistics Telephony <u>W</u> ir	eless <u>T</u> ools <u>H</u> elp	
	है 🚳 🤇 🗢 🔿 🖉 有	🛃 📃 🔍 Q	€ Ⅲ
Apply a display filter <ctrl-></ctrl->			
No. Time Source 4 1.001650690 0.0.0 5 1.003587619 10.0.105 6 108.135508801 YMware.4 7 109.137107830 YMware.4 9 111.185223948 YMware.4 10 112.209109349 YMware.4 11 113.233170076 YMware.4 13 140.7551499738 YMware.4 14 141.809150745 YMware.4 14 141.809150745 YMware.4 15 Frame 6: 42 Dytes on wire (3) Fthermet TI	Destination 255.255.255.255.255 11:00:36 Broadcast 11:00:36 Broadcast 11:00:36 Broadcast 11:00:36 Broadcast 11:00:36 Broadcast 11:00:36 Broadcast 11:00:36 Broadcast 11:00:36 Broadcast 11:00:36 Broadcast 11:00:36 Broadcast 11:00:37 (42) bytes captured (33 00:36 (40):270.41 (-276) Dett	Protocol Length Info DHCP 342 DHCP Req DHCP 342 DHCP ACK ARP 42 Who has ARP 42 Who has	uest - Transaction ID 0x7affa64a - Transaction ID 0x7affa64a 10.0255.1027 Tell 10.0.105.202 10.0.255.1027 Tell 10.0.105.202
Address Resolution Protocol Hardware type: Ethernet (: Protocol type: IPv4 (0x08) Hardware size: 6 Protocol size: 4 Opcode: request (1) Sender MAC address: VMwar Sender IP address: 10.0.1 Target IP address: 10.0.2	e_41:0c:36 (00:0c:29:41:0c:36) 05.202 100:00:00:00 (00:0c:29:41:0c:36) 05.202)	,
0000 ff ff ff ff ff ff 00 0c 0010 08 00 06 04 00 01 00 0c 0020 00 00 00 00 00 00 0a 00	29 41 0c 36 08 06 00 01 29 41 0c 36 0a 00 69 ca ff 66	·)A·6··· ·)A·6 <mark>··i·</mark> · · · f	

Wireshark Screen Layout

- When viewing a packet capture, the Wireshark screen is divided into three sections
- The bottom pane (packet bytes) shows the raw (binary) data associated with the packet selected in the packet list pane
 - If any decoded data is selected in the packet details pane the associated raw data will be highlighted in the packet bytes pane



Wireshark display filters can be typed into the filter toolbar to limit the data displayed and make it easier to view only specific packets

	*manufact-zone	
<u>File Edit View Go Capture Analyze</u>	<u>S</u> tatistics Telephony <u>W</u> ireless <u>T</u> ools <u>H</u> elp	
	९ 🗭 🔿 警 🗿 👱 🥃 📃 🔍 ९, ९, 🏦	
icmp		\boxtimes
No. Ime Fource 28 337.305579883 10.0.105.002 29 337.305579883 10.0.255.102 30 338.321174239 10.0.105.202 31 338.321174239 10.0.105.202 32 339.345141864 10.0.255.102 32 339.345141864 10.0.105.202 4 7 7 5 Frame 28: 98 bytes on wire (784 bit 6 Internet II, Src: VMware_41:0c:36 (r) 7 Internet Control Message Protocol	Destination Protocol Lengtr into 10.0.255.102 ICMP 98 Echo (ping) request id=0 10.0.255.102 ICMP 98 Echo (ping) reply id=0 10.0.105.202 ICMP 98 Echo (ping) request id=0 10.0.105.202 ICMP 98 Echo (ping) request id=0 10.0.205.202 ICMP 98 Echo (ping) request id=0 10.0.255.102 ICMP 98 Echo (ping) request id=0 98 bytes captured (784 bits) on interface manufact-zone, id 0 00:0c:29:41:0c:36), Dst: VMware_44:b2:9c (00:0c:29:44:b2:9c) 0.0.105.202, Dst: 10.0.255.102 00:0c:29:44:b2:9c) 00:0c:29:44:b2:9c)	<pre>bx64a7, seq=1/256, ttl=64 (reply in 29) bx64a7, seq=1/256, ttl=63 (request in 28) bx64a7, seq=2/512, ttl=64 (reply in 31) bx64a7, seq=2/512, ttl=63 (request in 30) bx64a7 sen=3/768 ttl=64 (renly in 33)</pre>
0000 00 0c 29 44 b2 9c 00 cc 29 41 00 0011 00 54 d1 aa 40 00 40 01 eb cd 00 0012 ff 66 08 00 c1 f5 64 a7 00 01 2 0030 00 00 00 00 00 00 00 00 00 00 01 10 14 16 17 18 19 1a 1b 1c 1d 1e 1f 24 26 27 28 29 2a 2b 2c 2d 2e 2f 36 0060 36 37 37 37 37 37 37 37	$\begin{array}{cccccccccccccccccccccccccccccccccccc$	

Here are some example and commonly used display filters

Filter	Purpose
Protocol name (i.e. dhcp, icmp, telnet)	Display only data from packets which implement a specific protocol
ip.addr == 192.168.1.1	Display only data coming from or going to the IP address 192.168.1.1
ip.src == 10.0.255.10 and ip.dst == 10.0.105.202	Display only data coming from the IP address 10.0.255.10 AND going to the IP address 10.0.105.202

For more information see <u>https://wiki.wireshark.org/DisplayFilters</u>

- Wireshark also supports capture filters which can be applied prior to starting the data capture
- Wireshark capture filters limit the data before capture while display filters limit the amount data display after capture

.using this filter: 📕 host 10.0.255.102		× 🛛	All interfaces shown *
manufact-zone Loopback: lo any bluetooth-monitor nflog nfqueue cell-area-zone Cisco remote capture: ciscodump DisplayPort AUX channel monitor capture: dpauxmon Random packet generator: randpkt systemd Journal Export: sdjournal SSH remote capture: sshdump	Λ.		
.earn Jser's Guide · Wiki · Questions and Answers · M	lailing Lists		

- Wireshark capture filters use a different syntax then display filters
- Wireshark capture filters use the pcap-filter syntax which is used by other network monitoring software packages such as the command line tcpdump program found on many Linux and Unix based systems
- For more information see <u>https://wiki.wireshark.org/CaptureFilters</u>

Filter Examples:

- Host: host 192.168.1.2
- HTTP: tcp and port 80
- Traffic between hosts: ip host 192.168.1.1 and 192.168.1.2
- Traffic from an host to another: *ip src 192.168.1.1 and dst 192.168.1.2*

Wireshark Follow Stream

Wireshark has the capability to combine all the packets in a protocol stream together then display them on a single screen in several different formats



Sy	/stem	ru	nin	g or	10	.0.2	255.	102	you	are	cor	nect	ing	from	10.	9.10	5.2	92.						
0 0	lient nk	ts. 1	serv	er ok	t. 0 ti	ums.																		
Er	tire c	0.014	orea	tion	(60	bute	(c)							ch		ad ca		data i	. 7	sci		Ctre		0 -
-	itire c	UIIV	c13a	cion	(05	byte	5)							311	Jw ai	iu se	ive	Jaca	15 1	USCI	·	Jan	am	0 -
Fin	d:																						Find	Next
														_										

Wireshark Packet Files

- Wireshark has the ability open, decode and analyze data saved in a wide variety of formats for example:
 - Wireshark's native format is libpcap which can be generated by many programs and network devices
 - Microsoft Network Monitor captures
 - Oracle snoop and atmsnoop captures
 - Novell LANalyzer captures
 - pppd log files
 - IBM OS/400 communication traces
 - MPEG-2 transport streams

Network interface cards (NIC) are designed to process network traffic addressed to themselves and discard all other network traffic



- To resolve this, some, but not all, network interface cards (NIC) can be configured to accept all traffic
 - Ethernet network cards may support promiscuous mode
 - Wireless network cards may support monitor mode



Network switches are designed to learn the addresses of systems connected to each port and store that information in a MAC address table



- Traffic is then forwarded out only on the port containing the system with the proper destination address
- Traffic from 192.168.1.101 to 192.168.1.109 would only be seen on ports 1 and 9



- There are multiple ways to resolve this, the following methods are often used by network administrators to legally monitor network traffic
 - Many network switches provide a feature that can be configured to mirror traffic from one another monitor port
 - Devices called network taps can be purchased and inserted into network where the tap will copy all traffic received onto a monitor port

- There are multiple ways to resolve this, the following methods are used by hackers to illegally monitor network traffic
 - Some switches MAC tables can be overloaded which will cause it to forward traffic out on all ports
 - A technique known as ARP spoofing can fool the switch into thinking a port contains an address it does not

For More Information

- For further information go to <u>https://www.nl.northweststate.edu/camo</u> or contact:
 - Tony Hills <u>thills@northweststate.edu</u> 419-267-1354
 - Sarah Stubblefield <u>sstubblefield@northweststate.edu</u> 419-267-1512
 - Mike Kwiatkowski <u>mkwiatkowski@northweststate.edu</u> 419-267-1231



Made possible through support from the National Science Foundation (NSF) award number <u>1800929</u>