

Capture and Display Filters

Editor's Note: This article is taken from Laura Chappell's book, Packet Filtering: Capturing the Cool Packets, which is available at www.podbooks.com.

A filter is a set of criteria that a packet must match to be accepted in the trace buffer or displayed in a protocol analyzer. For example, a filter based on all packets to 0xFF-FF-FF-FF-FF-FF is a standard broadcast filter.

As a protocol analyst, you must create a set of filters that match the traffic you are interested in viewing. Don't count on your protocol analyzer having a complete set of prebuilt filters. The vendors have been noticeably lax on supplying strong filters out of the box.

You need to know two things to build really great filters:

- The offset indicating where you are looking in the packet
- The value that you are looking for at that offset

When the packets come in off the wire, the protocol analyzer's card does some basic error checking on the packet. At this point, the packet is examined as a series of bytes with varying values.

The filters you build consist of an offset location and a value. If the incoming packet contains data that matches your filter in content and offset, the packet is said to match the filter. If a single bit does not match the filter value or offset, however, the packet is said to not match the filter.

Filters can be based on a number of packet characteristics—such as the source or destination hardware address (the media access control, or MAC, address), a single-bit setting in a flag field, or a specific ASCII character sequence in the data portion of the packet. As you would expect, however, the capabilities of protocol analyzers vary. They may not offer all of the options and filter types.

There are two types of filters: capture filters and display filters (also referred to as pre-filters and post-filters respectively).

CAPTURE FILTERS VERSUS DISPLAY FILTERS

Capture filters are placed on incoming traffic to reduce the amount of traffic that flows into the trace buffer. Display filters are placed on traffic in the trace buffer so that you can view specific types of packets as a subset of the trace buffer. (The original trace buffer contents are not erased.)

When do you use capture filters, and when do you use display filters? I use capture filters when I know the type of traffic I am



looking for straight off the wire. For example, I always set up my protocol analyzer with a wonderful Internet Control Message Protocol (ICMP) filter and run it on my client's network for as long as possible. (For more information about ICMP, see *TCP/IP Analysis and Troubleshooting* at www.podbooks.com. Alternately, consider getting the "Packet-Level ICMP" video course from www.podbooks.com. You'll see me ramble on passionately about ICMP for over an hour!) I may also build a filter for all traffic that meets my Gnutella or Morpheus filter to see if any ugly peer-to-peer applications are running across the network.

I always start my onsite visits with what I call the "laying on of hands"—the process of capturing all traffic on the network to get a general feel for the personality of a network. During this process, I scan through the summary window, looking for an interesting pattern or packet. If I find an interesting pattern, I apply display filters to view a series of subsets of the traffic. For example, if I am cruising through the summary and I see some unanswered Address Resolution Protocol (ARP) packets, I apply an ARP filter to the trace buffer and view only ARP traffic.

Note. Be careful here! If your protocol analyzer is attached to the network near a switch, you might be seeing only the outgoing ARP broadcasts. Unless the switch is set up for port spanning or mirroring, it will filter out the direct unicast ARP responses. Check to see if you have any other unicast packets. If you see only broadcast and multicast traffic, then unanswered traffic is the norm. Set up some port spanning or mirroring, or move over to a hubbed section of the network.

Next, I check to see if only one device is sending ARP packets that went unanswered or if hosts are searching unsuccessfully on the network. If only one host is sending out repetitive ARP packets looking up a single IP address, the source device may have a configuration fault, or a buggy piece of software (such as a printer driver that looks for specific print servers at specific locations) may be causing the problem.

If a whole bunch of hosts are sending unanswered requests for a single IP address, the target device may be down, or a slew of

hosts may be misconfigured. This problem may occur if a Dynamic Host Configuration Protocol (DHCP) server is configured incorrectly and messes up the clients.

Alternately, if one host is sending ARP packets for a whole series of IP addresses, I may investigate the sending host a bit further: Is the host performing some type of discovery for active hosts on the local network? Is the host a management agent that is trying to build a map of active hosts on a specific network segment? Is the host simply stupid?

Figure 1 shows a whole slew of ARPs that indicate a possible problem on the network. I mean, what is that guy doing in Figure 1? When does a system need to send out repeated ARP packets for various IP addresses? I would check out this station by filtering on all its traffic. This behavior is very weird! This may be a hacker looking around the network, or it may be a network management system building a network map.

WHEN YOU NEED BOTH

Sometimes you will need to use a combination of both capture and display filters: You could set up a broad filter (such as all broadcasts) first and then apply more specific filters (such as all ARP broadcasts). Or, you could set up a filter for all TCP/IP traffic and then, based on what you see there, apply filters for one user's traffic or one specific protocol.

Look at the traffic show in Figure 2. What traffic would you focus on next?

If you said, "the traffic from 127.0.0.1," you are right. Yipes! That's the loopback address! You should never see traffic to or from the loopback address out on the wire—that's internal-only traffic. You should first focus on all the traffic that uses 127.0.0.1. You may also want to look at any other traffic that used the same hardware address (hoping that it was originated from the local network).

In this case, a check on port 2301 indicated that this traffic was sent to/from Compaq's Insight Manager program—I'd call that a really ugly bug.

Now that I've defined when I apply capture filters and display filters, let me confuse you a bit. When using Network Associates' Sniffer, I never build display filters (by right-clicking on a trace file and selecting Define Filter). I only build capture filters (by clicking on the magic wand on the menu bar). I use these capture filters as both capture and display filters.

No.	Status	Source Address	Dest Address	Summary
1	U	0001963C3FA8	Broadcast	ARP: C PA=[12.234.1
2		0001963C3FA8	Broadcast	ARP: C PA=[12.234.1
3		0001963C3FA8	Broadcast	ARP: C PA=[12.234.1
4		0001963C3FA8	Broadcast	ARP: C PA=[12.234.1
5		0001963C3FA8	Broadcast	ARP: C PA=[12.234.1
6		0001963C3FA8	Broadcast	ARP: C PA=[12.234.1
7		0001963C3FA8	Broadcast	ARP: C PA=[10.75.7
8		0001963C3FA8	Broadcast	ARP: C PA=[12.234.1
9		0001963C3FA8	Broadcast	ARP: C PA=[12.234.1
10		0001963C3FA8	Broadcast	ARP: C PA=[12.234.1
11		0001963C3FA8	Broadcast	ARP: C PA=[10.109.1
12		0001963C3FA8	Broadcast	ARP: C PA=[12.234.1
13		0001963C3FA8	Broadcast	ARP: C PA=[12.234.1
14		0001963C3FA8	Broadcast	ARP: C PA=[24.254.1
15		0001963C3FA8	Broadcast	ARP: C PA=[10.109.1
16		0001963C3FA8	Broadcast	ARP: C PA=[12.234.1
17		0001963C3FA8	Broadcast	ARP: C PA=[12.234.1

Figure 1. Wow! This guy is out of control. No one needs to ARP this often.

I don't use Sniffer's display filters because the Sniffer's display filters can only be used on captured packets (ones that are already in the trace buffer). What a pain! Imagine it: You capture some data and then build a really hot display filter for all NetWare File Open NCP calls. Because you created this filter as a display filter in Sniffer, you cannot set up the analyzer to capture only File Open NCP call packets directly off the wire.

Sniffer's display filters can be used only on traffic that has already been captured. Sniffer's capture filters can be used to grab specific traffic directly off the wire or reduce the number of packets within a trace file. I'm not sure why Network

Associates architected the product this way. (If I find out, I'll put a note out on www.packet-level.com.)

CONCLUSION

Building filters is the key to protocol analysis. If you understand how to build filters based on specific packet characteristics, you can isolate suspicious traffic and track down the cause.

Laura Chappell performs onsite network analysis sessions for troubleshooting, optimization, and security checks. She teaches hands-on courses on protocol analysis. For more information about how she can help your company's network run more efficiently and more securely, visit www.packet-level.com. ●

No.	S	Source Address	Dest Address	Summary
172		0060FD8B70FA	Bridge_Group_Add	BFDU: S Pri=8000 F
172		0060FD8B70FA	Bridge_Group_Add	BFDU: S Pri=8000 F
172		[0.0.0.0]	[255.255.255.255]	DHCP: Request, Mes
172		[0.0.0.0]	[255.255.255.255]	DHCP: Request, Mes
172		00D05937C9BF	Broadcast	ARP: C PA=[10.30.0
172		00D05937C9BF	Broadcast	ARP: C PA=[10.30.0
172		0060FD8B70FA	Bridge_Group_Add	BFDU: S Pri=8000 F
172		00D05937C9BF	Broadcast	ARP: C PA=[10.30.0
172		[127.0.0.1]	[255.255.255.255]	UDP: D=2301 S=2301
173		0060FD8B70FA	Bridge_Group_Add	BFDU: S Pri=8000 F
173		101433C.0001E	101433C.FFFFFFFF	
173		0060FD8B70FA	Bridge_Group_Add	BFDU: S Pri=8000 F
173		00E01659824A	010081000100	BTDP: Type=Flatnet
173		00E01659824A	010081000101	BTDP: Type=Segment
173		[127.0.0.1]	[255.255.255.255]	UDP: D=2301 S=2301
173		[127.0.0.1]	[255.255.255.255]	UDP: D=2301 S=2301
173		[10.30.0.34]	[255.255.255.255]	UDP: D=2301 S=2301
173		0060FD8B70FA	Bridge_Group_Add	BFDU: S Pri=8000 F
173		0060FD8B70FA	Bridge_Group_Add	BFDU: S Pri=8000 F

Figure 2. Which traffic would you focus on next?