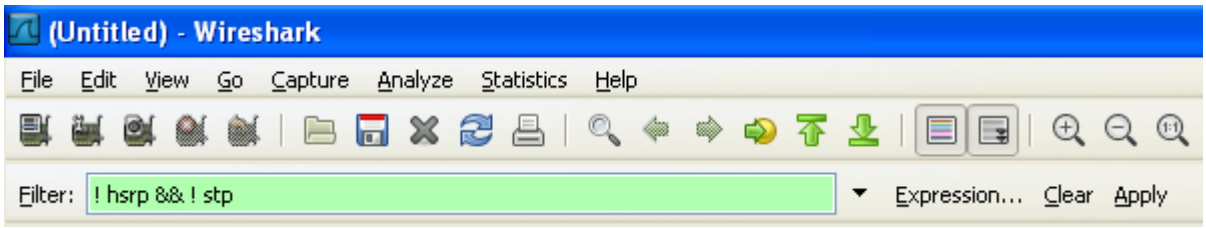
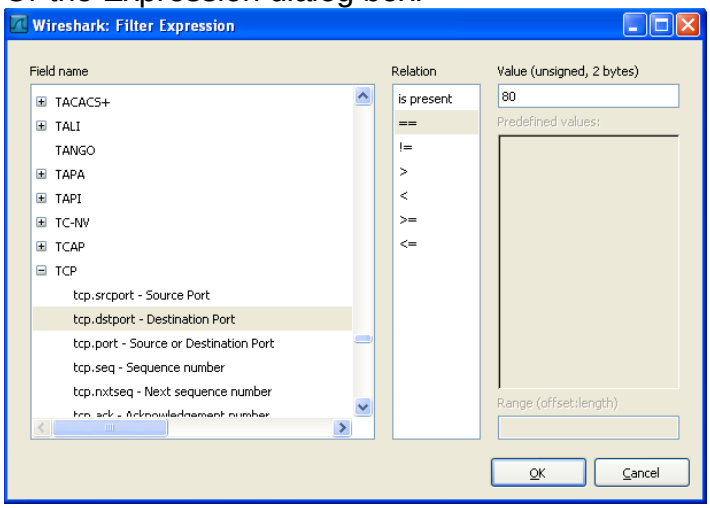
Creating WireShark Filters

You can use the filter bar:



NOTE: The expression bar turns green when a filter rule is understood

Or the Expression dialog box:



Filter Commands:

NOTE: WireShark is case sensitive use all lower case in filters

Operator	Description / Example
`&&` And	`http && tcp.port==2120` Shows HTTP packets AND further restricts the filter to only showing traffic to or from tcp port 2120 (aka random browser port)
`||` Or	`dns || http` Show both DNS and HTTP packets *NOTE: && would not work here as a packet can't be both DNS and HTTP at the same time.*
`!` Not	`!hsrp && !stp` Shows all traffic except HSRP and STP frames
`==` Equal	`eth.src==00:04:75:af:04:9c` Shows all frames with a source Ethernet address of 00:04:75:af:04:9c
`!=` Not Equal	`eth.src!=00:04:75:af:04:9c` Shows all frames that DID NOT have a source Ethernet address of 00:04:75:af:04:9c
`>` Greater Than	`udp.srcport > 1024` Shows all UDP segments who's source port number is greater than 1024
`<` Less Than	`tcp.srcport < 1024` Shows all tcp segments who's source port number is less than 1024

>= Greater than or equal to	udp.srcport >= 1024 Shows all UDP segments who's source port number is greater than or equal to 1024
<= Less than or equal to	tcp.dstport <= 1024 Shows all TCP segments who's source port number is below or equal to 1024
( ) Grouping	(arp || dns) && ! ip.src==192.168.40.44 The () in this examples ensure that only arp OR DNS packets are displayed AND then excludes traffic that has a source address of 192.168.40.44
eth.addr eth.src eth.dst	eth.addr==00:04:75:af:04:9c Used with one of the other operators to show to and from MAC addresses
ip.addr Ip.src Ip.dst	Ip.src ==192.168.1.1 Used with one of the other operators to show packets from the IP addresse 192.168.1.1
tcp.port tcp.srcport tcp.dstport	Tcp.port==80 Used with one of the other operators to show segments to or from TCP port 80

For more examples see: http://wiki.wireshark.org/DisplayFilters.

WireShark filter approaches:

	Pros	Cons
Explicitly show packets exclude everything else. Example: arp && dns	• Shows only the specific packets, reducing the need to look through large amounts of information • Easier to spot trends over time	• Can be difficult to write a filter that includes the entire communication • May eliminate important lower layer protocols that are significant to the communication
Filter specific extraneous packets, and include everything else Example: !hsrp && !stp	• Easier to create • More likely to see the entire communication • Uses a build as you go approach	• Can still produce significant numbers of packets to inspect