

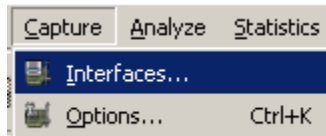
Wireshark Hands-On Exercises

Step 1. Plug in the Airpcap USB device.

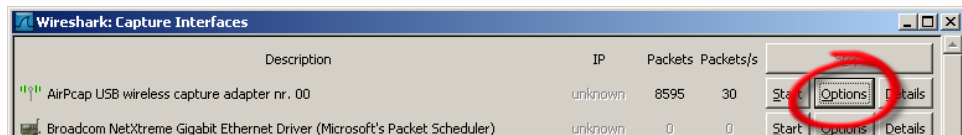


Step 2. Open Wireshark - Start → Wireless Tools → Wireshark.

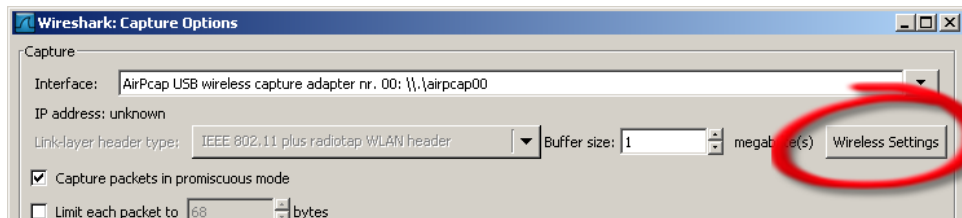
Step 3. Click on Capture → Interfaces.



Step 4. Choose the AirPcap USB adapter and click on Options to set details for this capture.



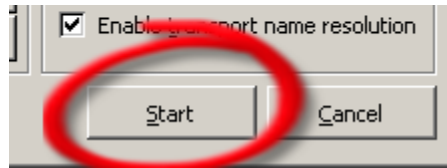
Step 5. Review the options on this page... then click on Wireless Settings.



Step 6. Select **channel 1** as the channel we'll be capturing from.



Step 7. Return to the Options page, then click **Start** button to start your capture.



Step 8. Note, right now all packets are being shown as they come to the wireless card.

Step 9. Review the notes below on how to make and use Filters in Wireshark.

Step 10. Create a Filter to display all traffic except beacons.

Filter: `!wlan.fc.subtype==8`

Step 11. Create a Filter to display only data traffic.

Filter: `wlan.fc.type==2`

Step 12. Create a Filter to display only Data... but NOT NULL Data (going to sleep) packets.

Filter: `wlan.fc.type==2 and !(wlan.fc.type_subtype == 36)`

Step 13. Now try some new filters on your own.

NOTE: You can review more on Wireshark from the Laura Chappell's new Wireshark Study Guide Book.

Step 14. Create a Filter to capture only voice traffic.

Step 15. Create a Filter to capture only FTP traffic.

Step 16. Create a Filter to capture only traffic to a destination network.

Step 17. Create a Filter to capture only traffic to a destination host.

Step 18. How about a filter to capture Access Points with 'cloaked' or 'hidden' SSIDs? When an Access Point does NOT broadcast SSID, the SSID field contains no data in Beacons and Probe Response packets. But... clients MUST ask for the proper 'hidden' SSID in their requests to join the BSA.

NOTE: This filter is `wlan.bssid==xx:xx:xx:xx:xx:xx and wlan.fc.type_subtype==0` where the BSSID of the Access Point you are looking for is in the xx's.

By applying the above filter, we reveal any association requests for the specific BSSID. By clicking [IEEE 802.11 Wireless LAN Management Frame → Tagged Parameters → SSID Parameter Set](#) in the packet detail window we can see the SSID requested by the client station, thus revealing the ‘Hidden’ SSID.

Wireshark Filters for 802.11 Frames

802.11 Header Field

| | |
|--------------------------------------|---------------|
| Either Source or Destination Address | wlan.addr |
| Transmitter Address | wlan.ta |
| Source Address | wlan.sa |
| Receiver Address | wlan.ra |
| Destination Address | wlan.da |
| BSSID | wlan.bssid |
| Duration | wlan.duration |

Frame Control Subfields

| | |
|----------------------------|-----------------|
| Frame Type | wlan.fc.type |
| Frame Subtype | wlan.fc.subtype |
| ToDS Flag | wlan.fc.tods |
| FromDS Flag | wlan.fc.fromds |
| Retry Flag | wlan.fc.retry |
| Protected Frame (WEP) Flag | wlan.fc.wep |

Fields can be combined using operators. Wireshark supports a standard set of comparison operators:

| | | | |
|----|------------------|----|------------------------------|
| == | for equality | != | for inequality |
| > | for greater than | >= | for greater than or equal to |
| < | for less than | <= | for less than or equal to |
| && | Contains | | Matches |
| ! | Not | | |

An example of a display filter would be `wlan.fc.type==1` to match control frames.

To remove all Beacon frames from your trace, you’ll need to write a display filter that matches Beacon frames, and then negate it. Like the example below:

- Filter on type code for management frames with `wlan.fc.type==0`
- Filter on subtype code for Beacon with `wlan.fc.subtype==8`

Combine the two, and negate the operation by using the exclamation point for NOT with an expression result of:

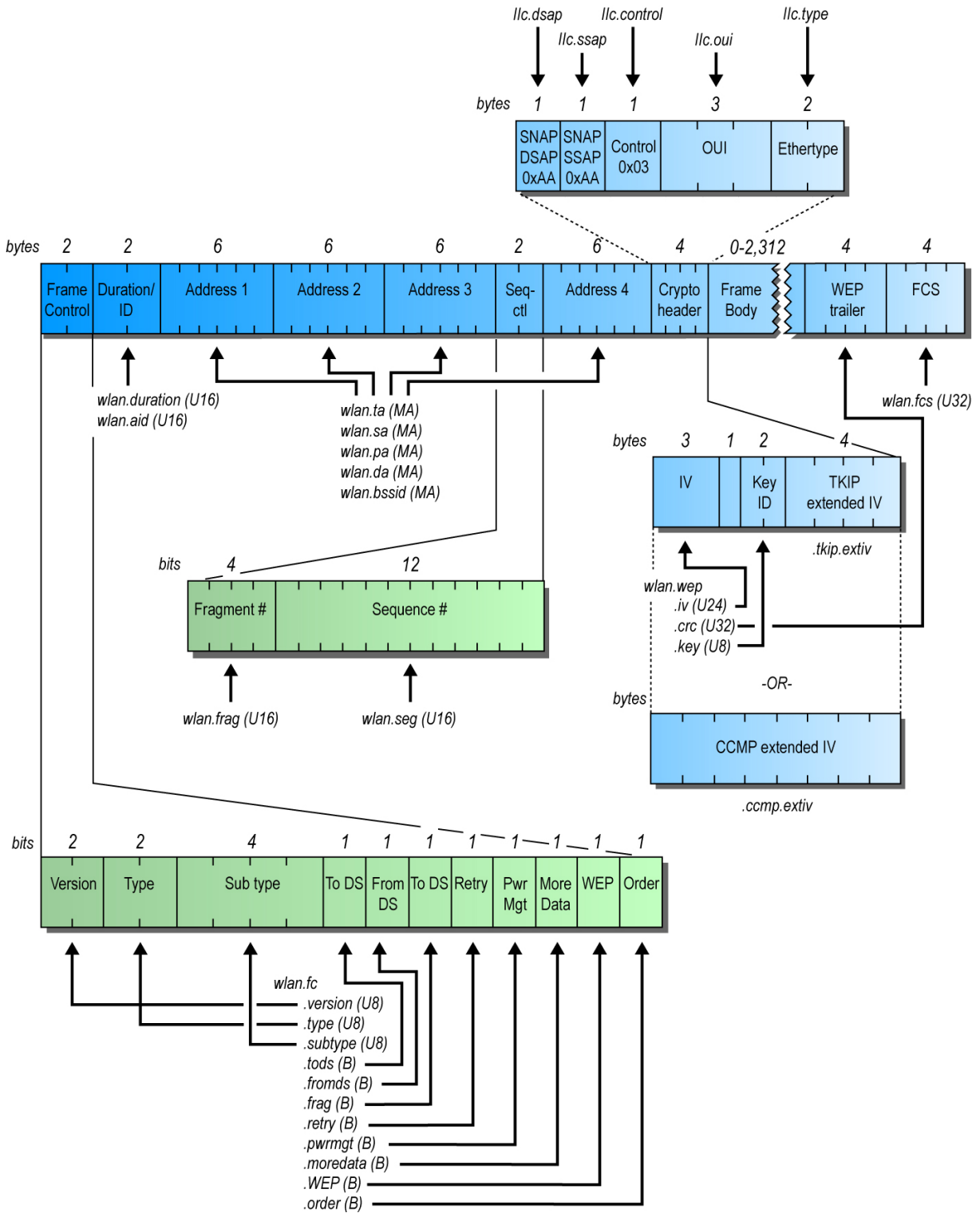
`! (wlan.fc.type==0 and wlan.fc.subtype==8)`

When assessing a wireless capture with Wireshark, it is common to apply display filters to look for or exclude certain frames based on the IEEE 802.11 frame type and frame subtype files. If you are trying to exclude frames from a capture, it is easy to identify the Type and Subtype filed by navigating the Packet Details windows and use those values for your filter.

Or, you can just use this handy-dandy table we've provided below.

| Frame Type/Subtype | Filter |
|-----------------------|---------------------------------------|
| Management Frames | <code>wlan.fc.type==0</code> |
| Association Request | <code>wlan.fc.type_subtype==0</code> |
| Association Response | <code>wlan.fc.type_subtype==1</code> |
| Ressociation Request | <code>wlan.fc.type_subtype==2</code> |
| Ressociation Response | <code>wlan.fc.type_subtype==3</code> |
| Probe Request | <code>wlan.fc.type_subtype==4</code> |
| Probe Response | <code>wlan.fc.type_subtype==5</code> |
| Beacon | <code>wlan.fc.type_subtype==8</code> |
| ATIM | <code>wlan.fc.type_subtype==9</code> |
| Disassociate | <code>wlan.fc.type_subtype==10</code> |
| Authentication | <code>wlan.fc.type_subtype==11</code> |
| Deauthentication | <code>wlan.fc.type_subtype==12</code> |
| Association Request | <code>wlan.fc.type_subtype==0</code> |
| Association Request | <code>wlan.fc.type_subtype==0</code> |
| Control Frames | <code>wlan.fc.type==1</code> |
| Power-Save Poll | <code>wlan.fc.type_subtype==26</code> |
| Request To Send - RTS | <code>wlan.fc.type_subtype==27</code> |
| Clear To Send - CTS | <code>wlan.fc.type_subtype==28</code> |
| Acknowledgement - ACK | <code>wlan.fc.type_subtype==29</code> |
| Data Frmaes | <code>wlan.fc.type==2</code> |
| NULL Data | <code>wlan.fc.type_subtype==36</code> |

Here is a great graphical view of Wireshark's 802.11 Filter names for each part of an 802.11 frame.



Display Filter Syntax

| | |
|-------------------|---|
| Hosts/Network | ip.addr, ip.src, ip.dst, eth.addr, eth.src, eth.dst |
| Ports | tcp.port, tcp.srcport, tcp.dstport, udp.port, udp.srcport, udp.dstport |
| Various Protocols | arp, bootp, dcerpc, dns, eth, ftp, http, icmp, ip, ncp, netbios, ntp, ospf, sip, smtp, snmp, tcp, udp |
| Examples | ip.addr==10.4.2.19 |
| | !ip.addr==10.4.15.27 |
| | !arp && !bootp |
| | tcp.port==80 |
| | eth.dst==00:04:5a:df:80:37 |
| | tcp.flags.reset==1 |

Keyboard Shortcuts

| | |
|---------------|---|
| Tab | Move forward between packet windows and screen elements |
| Shift-Tab | Move backwards between packets windows screen elements |
| Down | Move forward to the next packet or detail item |
| Up | Move back to the previous packet or detail item |
| Ctrl-Down, F8 | Move to the next packet, even if the packet list is not the focus. |
| Ctrl-Up, F7 | Move to the previous packet, even if the pack list is not the focus. |
| Left | Closes the selected tree item in the packet detail window or move to the parent node if already closed. |
| Right | Expands the selected tree item in the packet detail window (does not expand the subtree) |
| Backspace | Move to the parent node in the packet detail window |
| Return, Enter | Toggles expansion of the selected tree item in the packet detail window |
| Ctrl-M | Mark a packet |
| Ctrl-N | Go to the next market packet |
| Ctrl-T | Set time reference |
| Ctrl-Plus | Zoom in (increase font size) |
| Ctrl-Minus | Zoom out (decrease font size) |
| Ctrl-Equal | Zoom to 100% |