**EXPERIMENT NO. 7**

Aim: Use Wireshark to understand the operation of TCP/IP layers.

Requirements: Linux/Windows O.S, Compatible version of Wireshark.

Theory:

What is Wireshark?

Wireshark is a network packet analyzer. A network packet analyzer presents captured packet data in as much detail as possible.

You could think of a network packet analyzer as a measuring device for examining what’s happening inside a network cable, just like an electrician uses a voltmeter for examining what’s happening inside an electric cable (but at a higher level, of course).

In the past, such tools were either very expensive, proprietary, or both. However, with the advent of Wireshark, that has changed. Wireshark is available for free, is open source, and is one of the best packet analyzers available today.

Some intended purposes

Here are some reasons people use Wireshark:

* Network administrators use it to *troubleshoot network problems*
* Network security engineers use it to *examine security problems*
* QA engineers use it to *verify network applications*
* Developers use it to *debug protocol implementations*
* People use it to *learn network protocol* internals

Wireshark can also be helpful in many other situations.

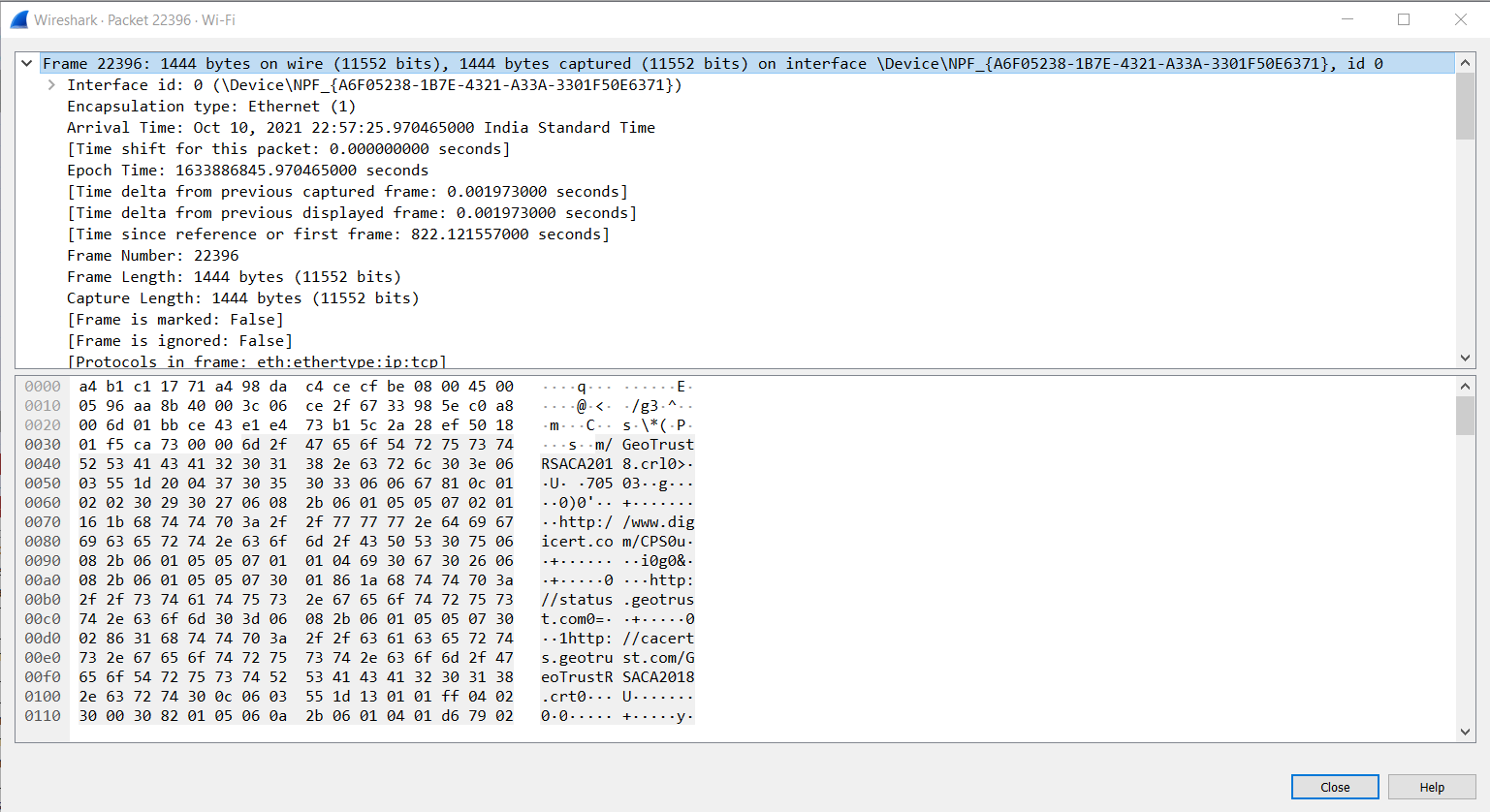
Features:

The following are some of the many features Wireshark provides:

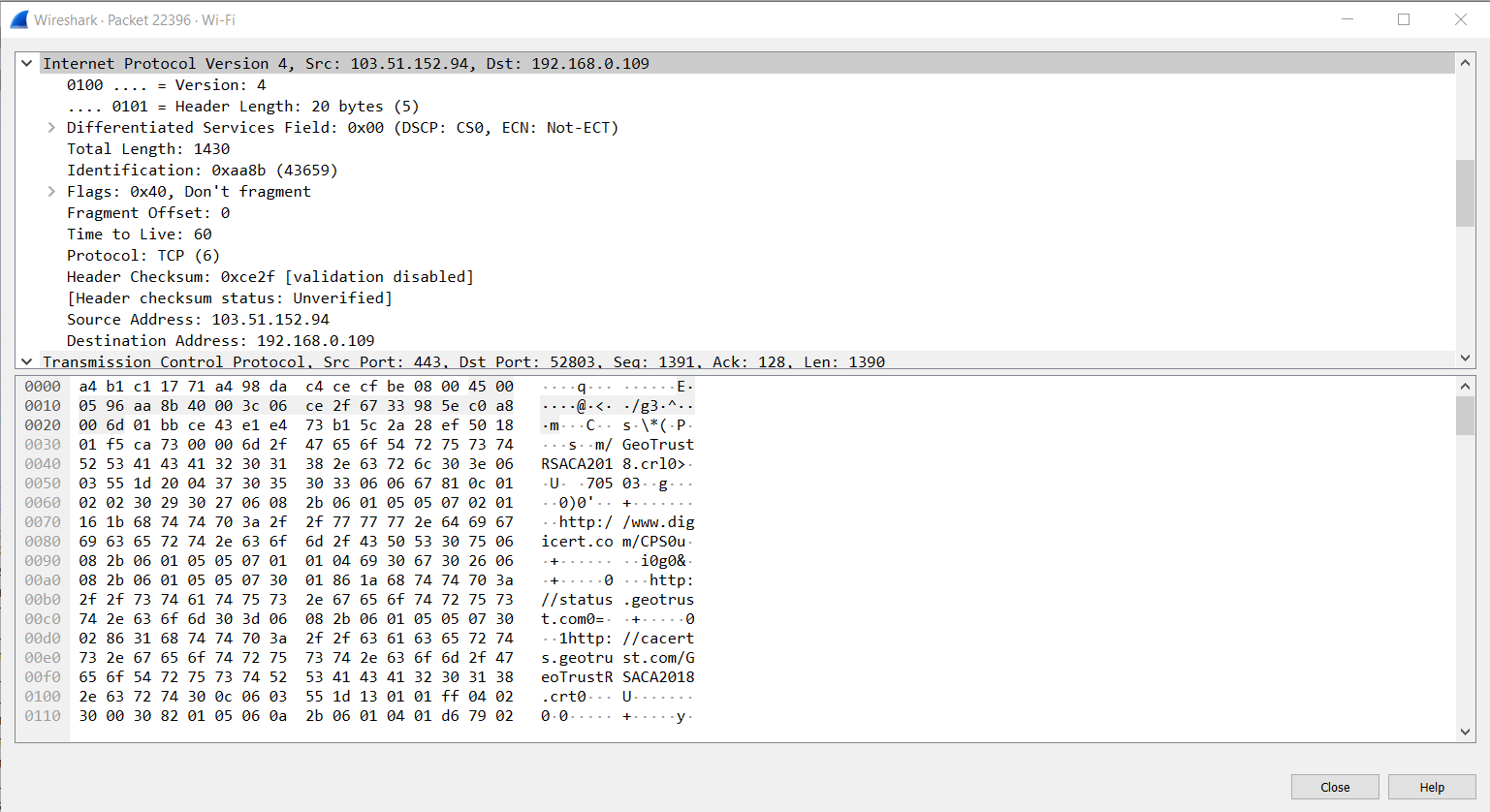
* Available for *UNIX* and *Windows*.
* *Capture* live packet data from a network interface.
* *Open* files containing packet data captured with tcpdump/WinDump, Wireshark, and many other packet capture programs.
* *Import* packets from text files containing hex dumps of packet data.
* Display packets with *very detailed protocol information*.
* *Save* packet data captured.
* *Export* some or all packets in a number of capture file formats.
* *Filter packets* on many criteria.
* *Search* for packets on many criteria.
* *Colorize* packet display based on filters.
* Create various *statistics*.
* …​And *a lot more!*

Wireshark Output:

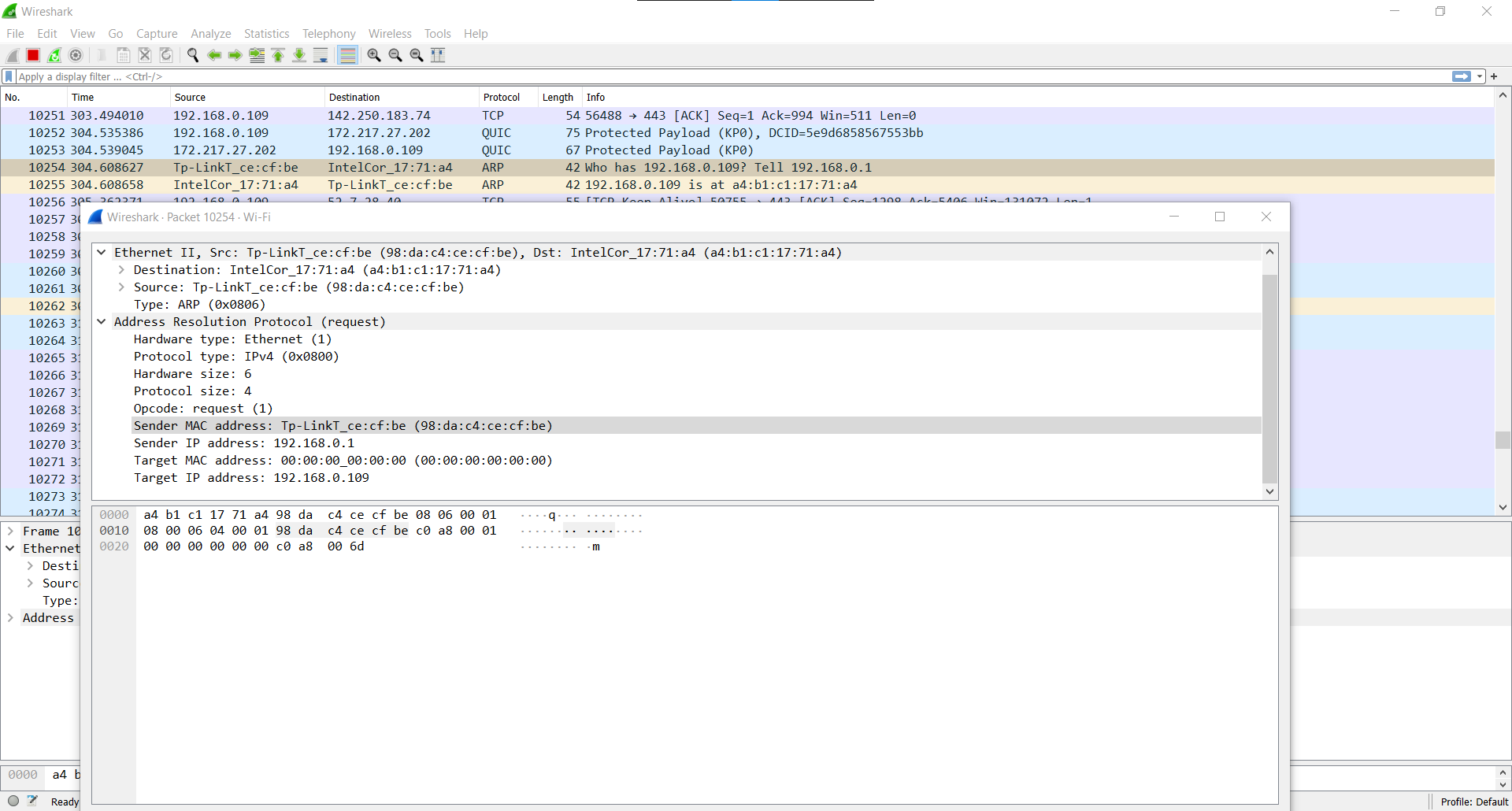
Frame Header and Frame Size:



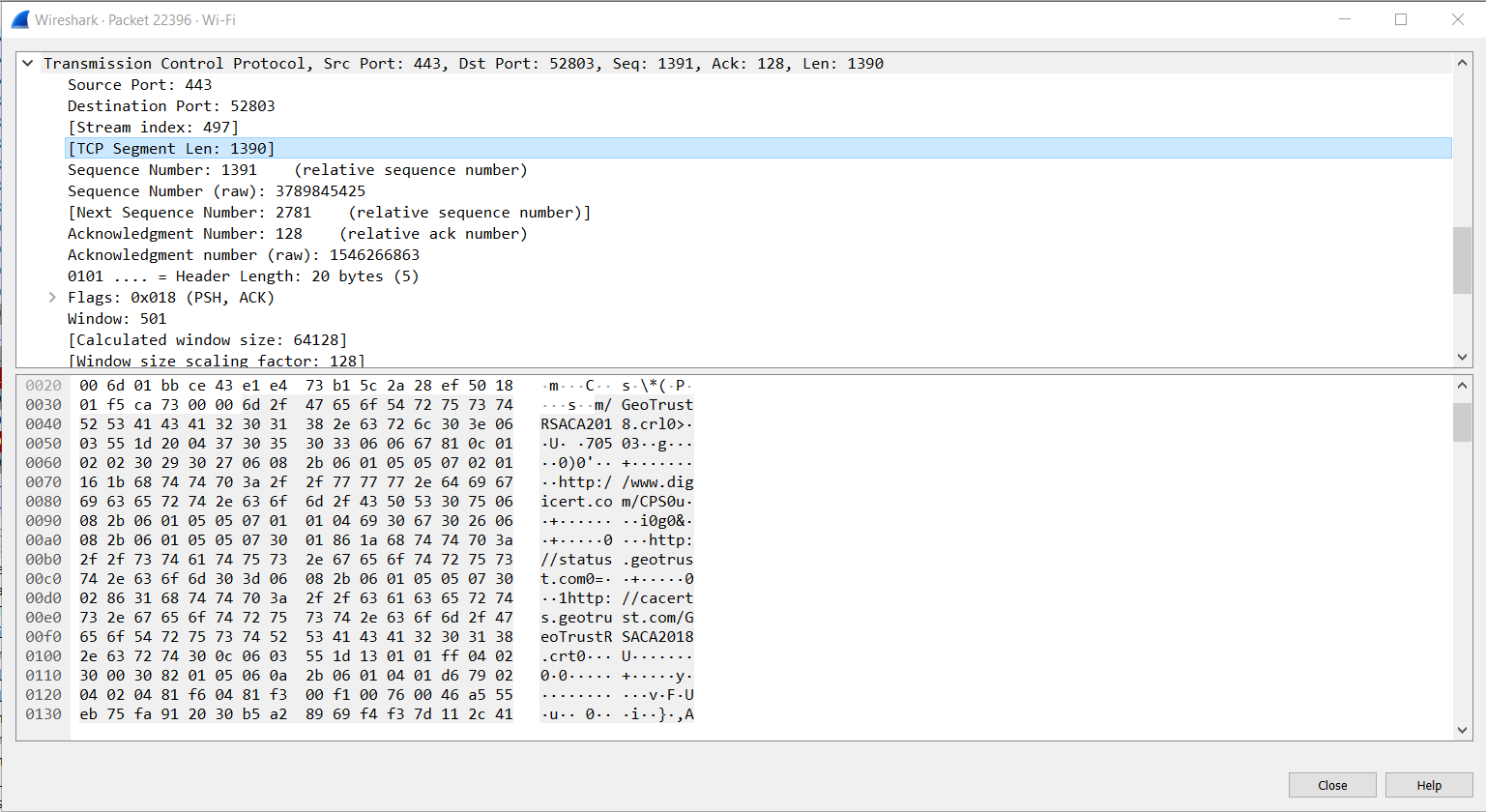
IP header:



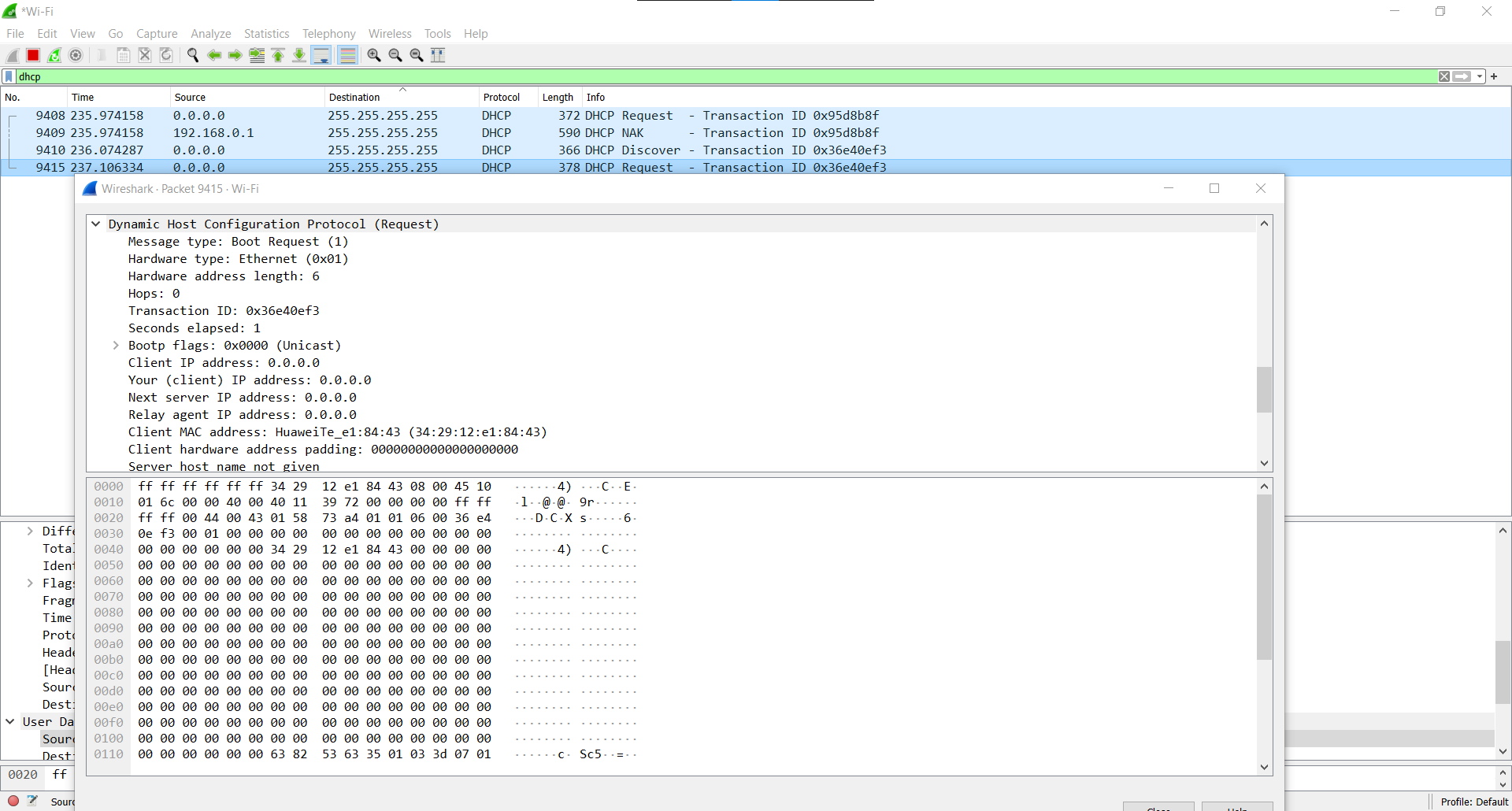
MAC Address and ARP:



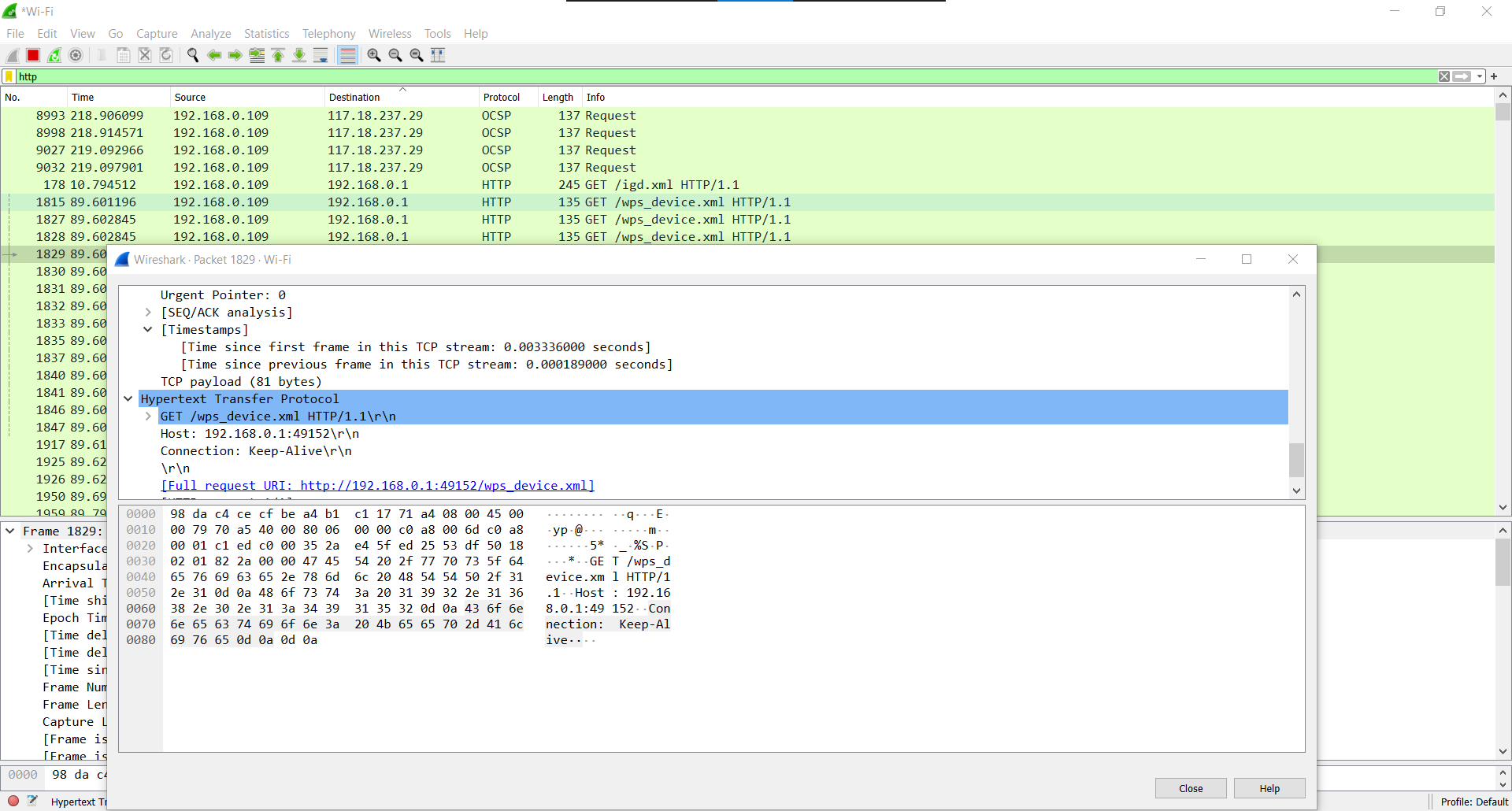
TCP Header:



DHCP:



HTTP:



Conclusion: We have successfully use Wireshark to understand the operation of TCP/IP layers and use it to get different header formats of packets.