Experiment No. 1

Aim: Monitoring the network traffic using Wireshark

2. Objectives: To observe the performance in promiscuous & non-promiscuous mode & to find the packets based on different filters.

3. Outcomes: The learner will be able to:-

Identify different packets moving in/out of network using packet sniffer for network analysis.

Understand professional, ethical, legal, security and social issues and responsibilities. Also will be able to analyze the local and global impact of computing on individuals, organizations, and society.

Match the industry requirements in the domains of Database management, Programming and Networking with the required management skills.

4. Hardware / Software Required: Wireshark, Ethereal and tcpdump.

5. Theory:
Wireshark, a network analysis tool formerly known as Ethereal, captures packets in real time and display them in human-readable format. Wireshark includes filters, color-coding and other features that let you dig deep into network traffic and inspect individual packets.
Applications:
Network administrators use it to troubleshoot network problems

Network security engineers use it to examine security problems

Developers use it to debug protocol implementations

People use it to learn network protocol internals beside these examples can be helpful in many other situations too. **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 a number of other packet capture programs.

Import packets from text files containing hex dumps of packet data.

Display packets with very detailed protocol information.

Export some or all packets in a number of capture file formats.

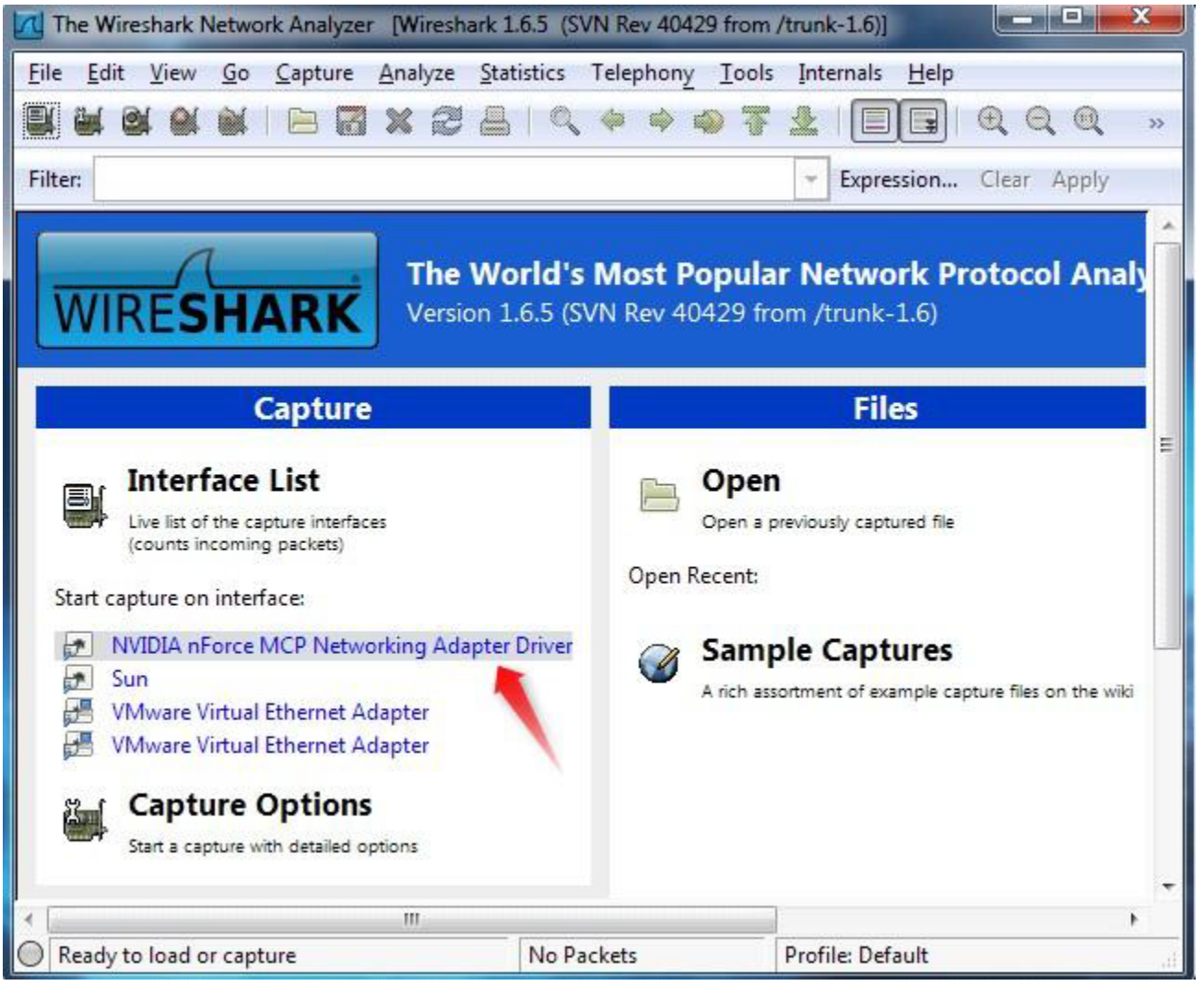
Filter packets on many criteria.

Search for packets on many criteria.

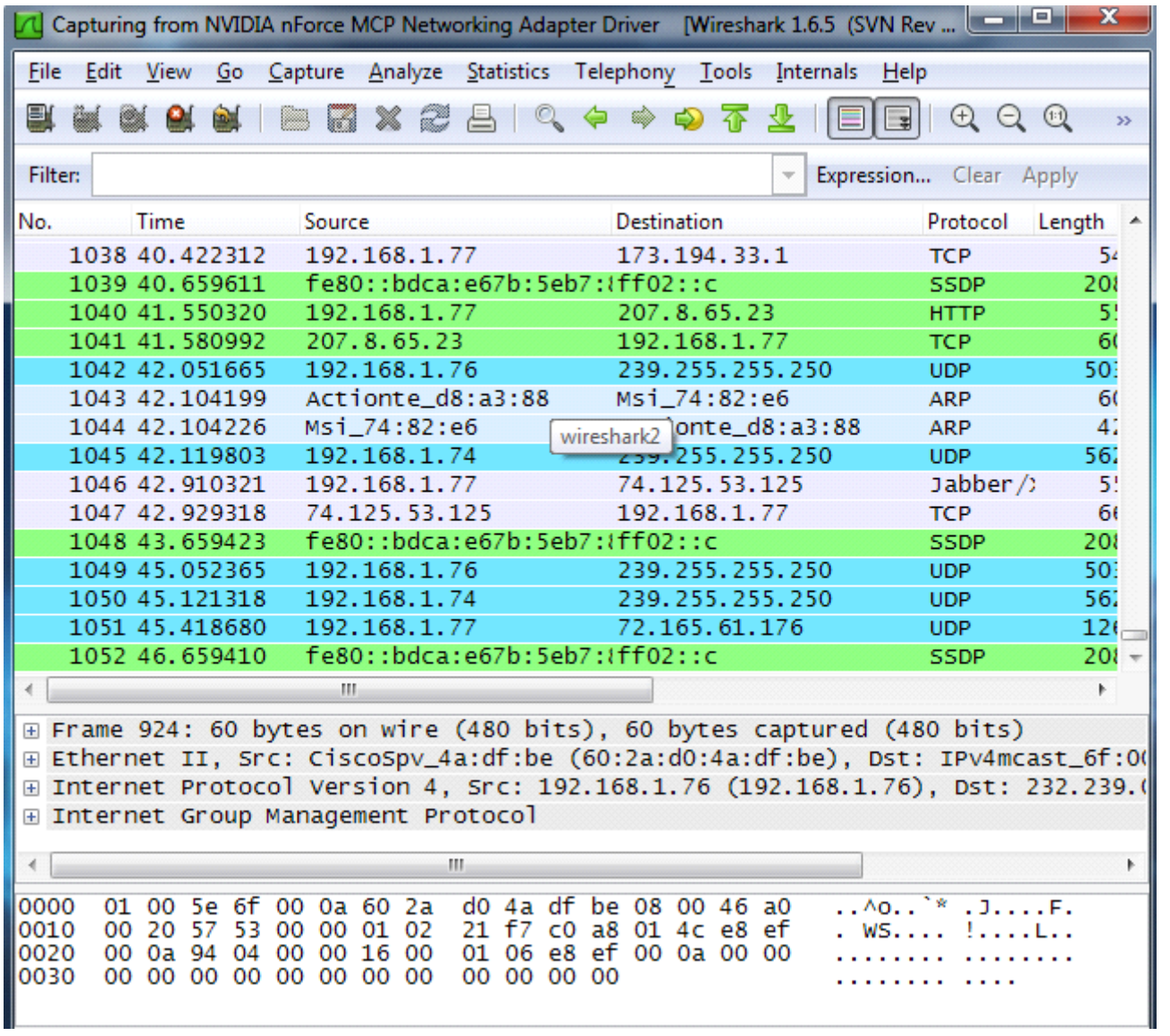
Colorize packet display based on filters.

Create various statistics.

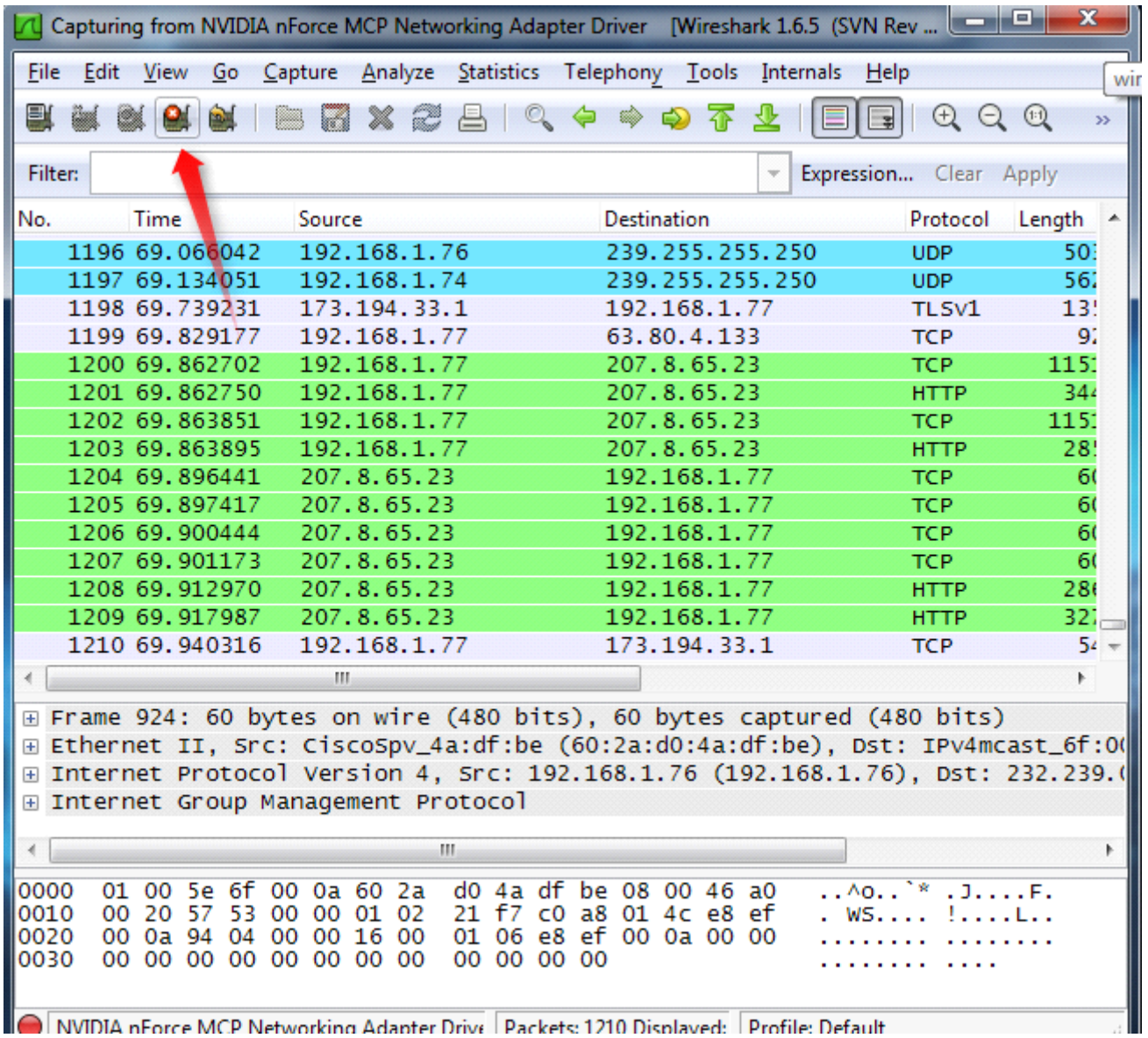
Capturing Packets
After downloading and installing wireshark, you can launch it and click the name of an interface under Interface List to start capturing packets on that interface. For example, if you want to capture traffic on the wireless network, click your wireless interface. You can configure advanced features by clicking Capture Options.



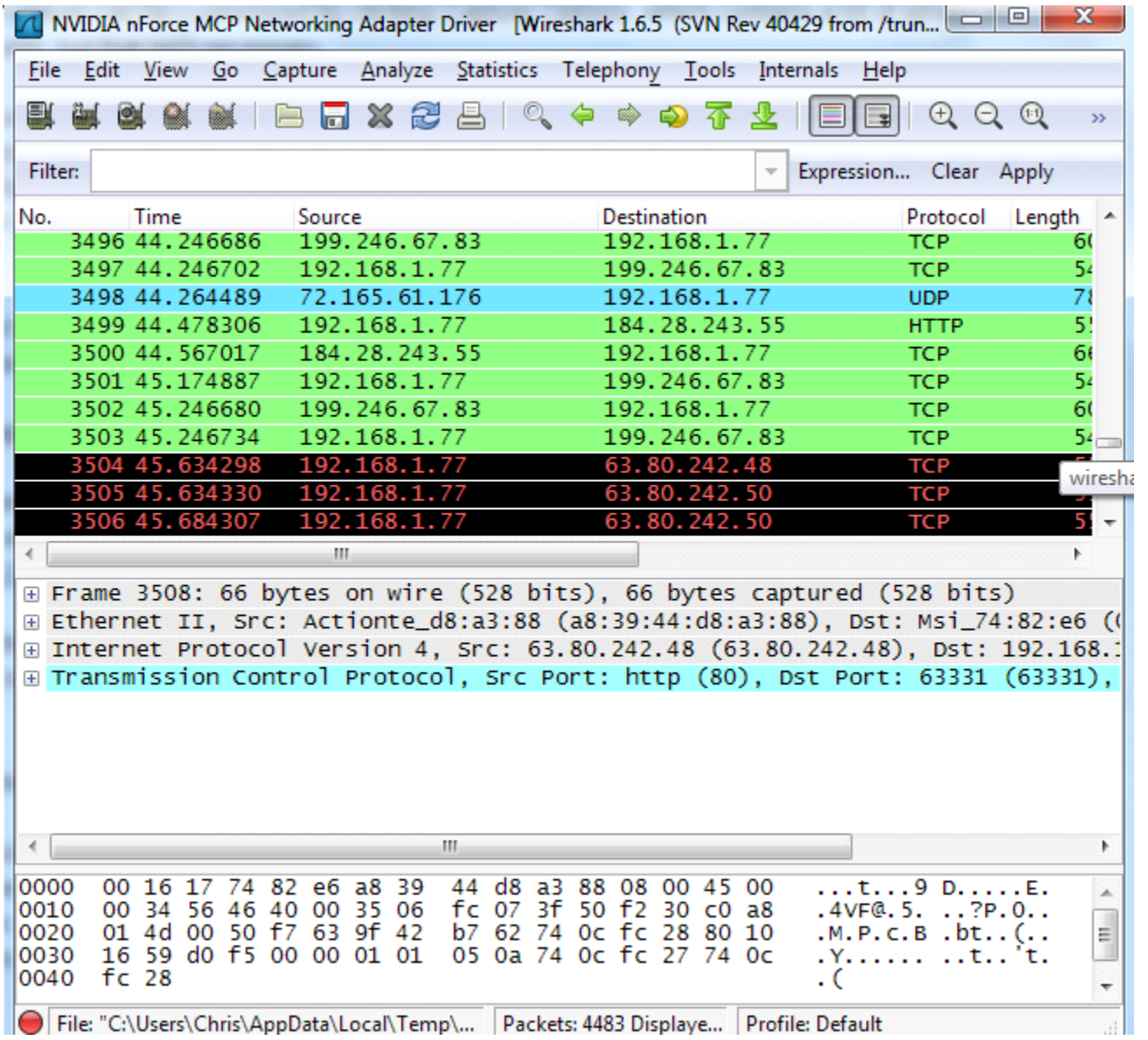
As soon as you click the interface's name, you'll see the packets start to appear in real time. Wireshark captures each packet sent to or from your system. If you're capturing on a wireless interface and have promiscuous mode enabled in your capture options, you'll also see other the other packets on the network



Click the stop capture button near the top left corner of the window when you want to stop capturing traffic.



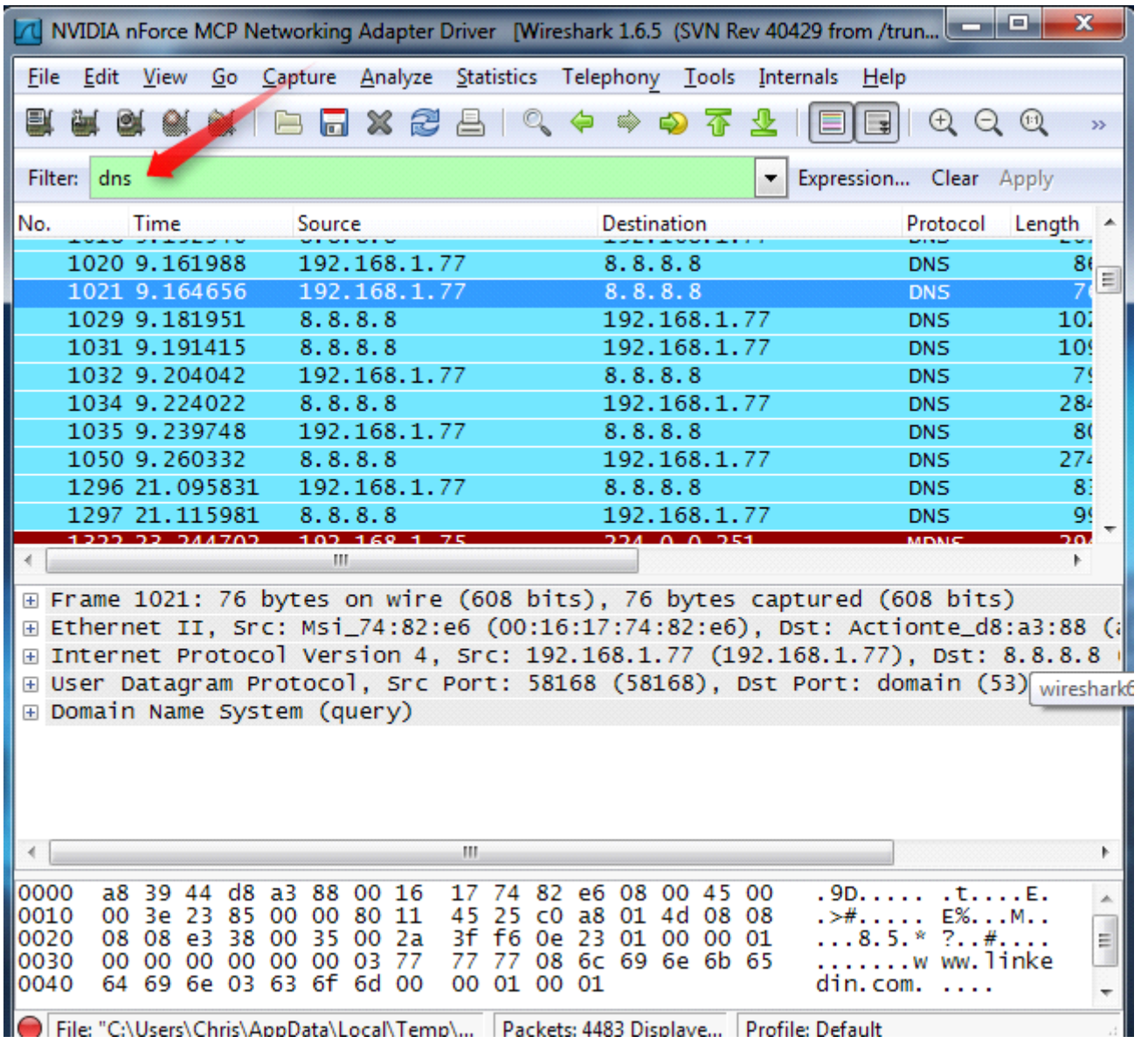
Wireshark uses colors to help you identify the types of traffic at a glance. By default, green is TCP traffic, dark blue is DNS traffic, light blue is UDP traffic, and black identifies TCP packets with problems — for example, they could have been delivered out-of-order.



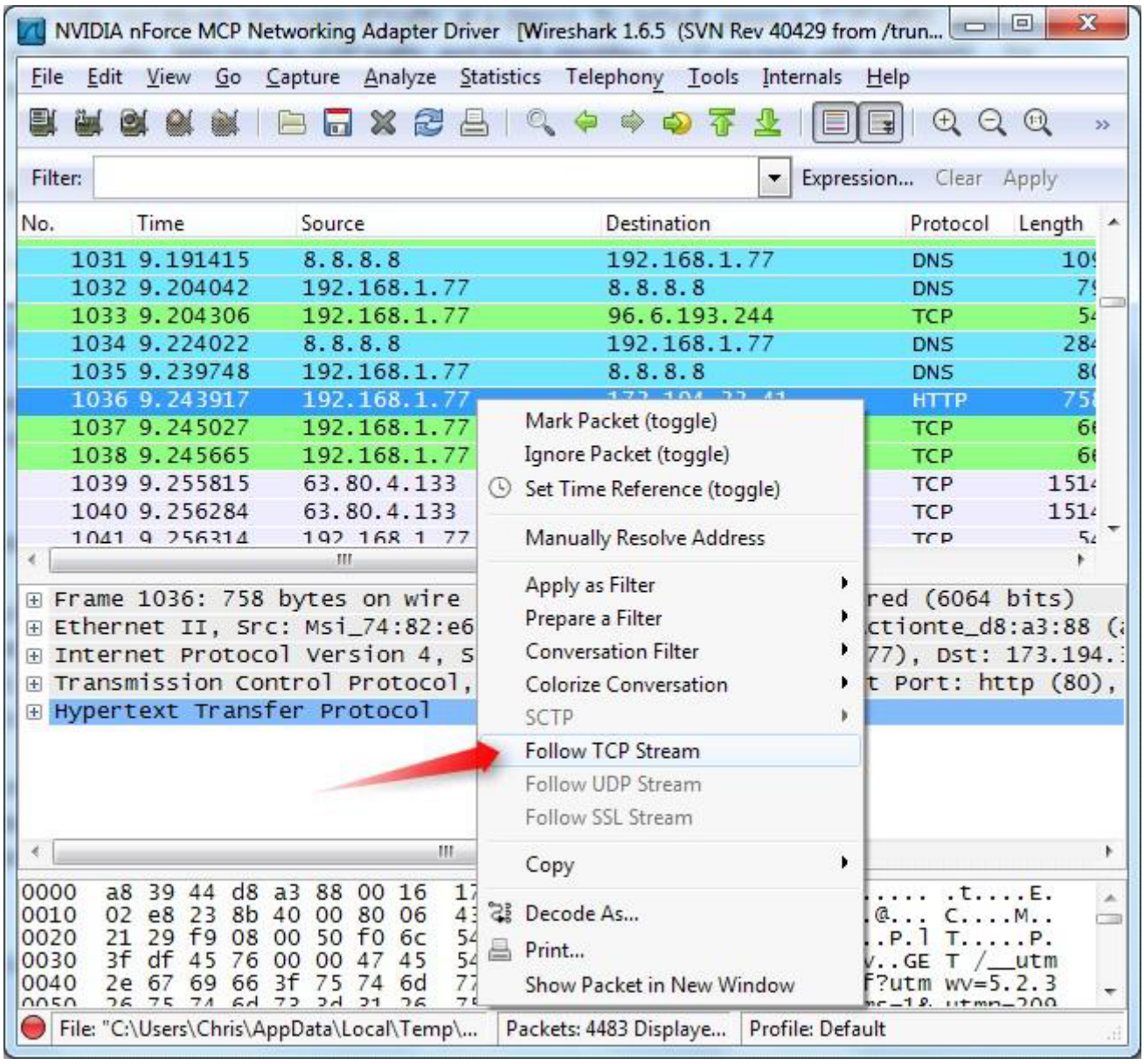
Filtering Packets

If you're trying to inspect something specific, such as the traffic a program sends when phoning home, it helps to close down all other applications using the network so you can narrow down the traffic. Still, you'll likely have a large amount of packets to sift through. That's where Wireshark's filters come in.

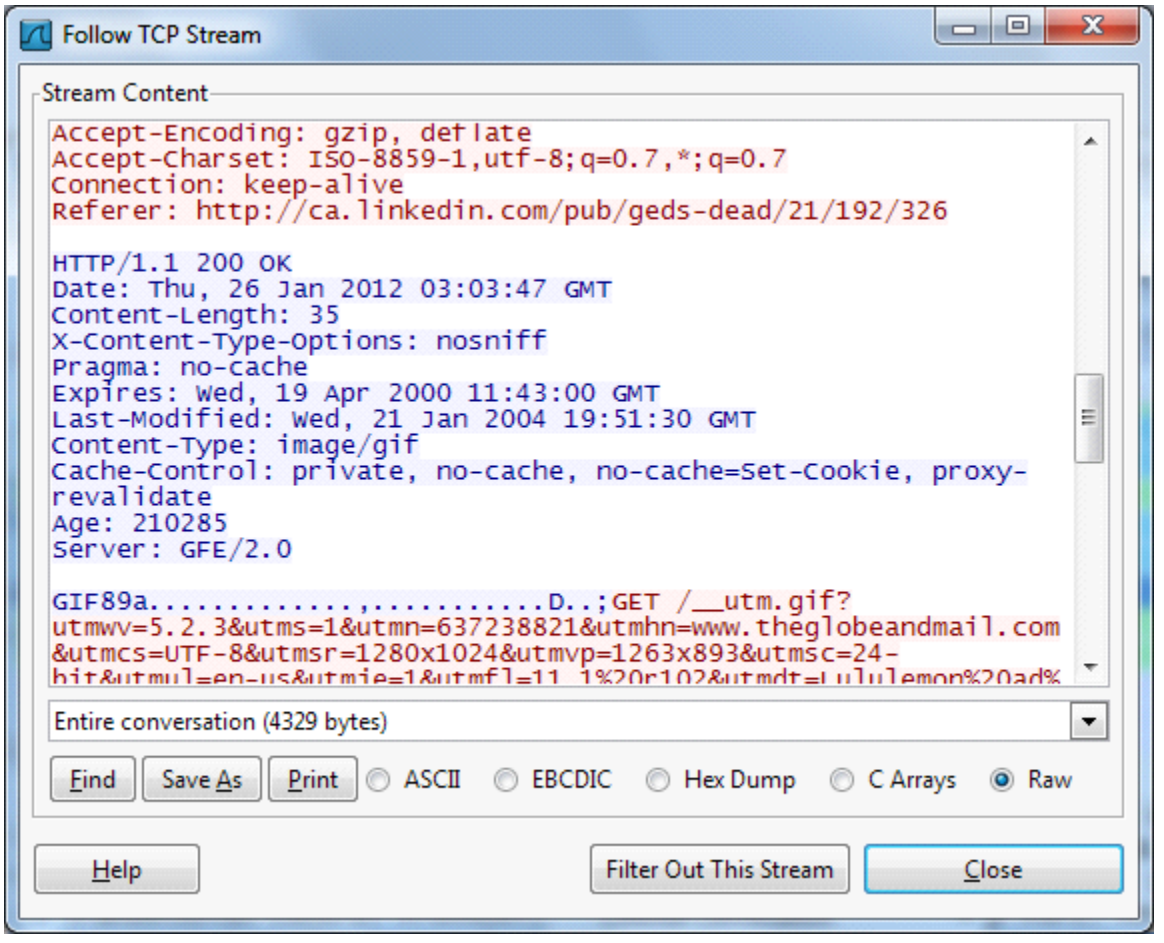
The most basic way to apply a filter is by typing it into the filter box at the top of the window and clicking Apply (or pressing Enter). For example, type ―dns‖ and you'll see only DNS packets. When you start typing, Wireshark will help you autocomplete your filter.



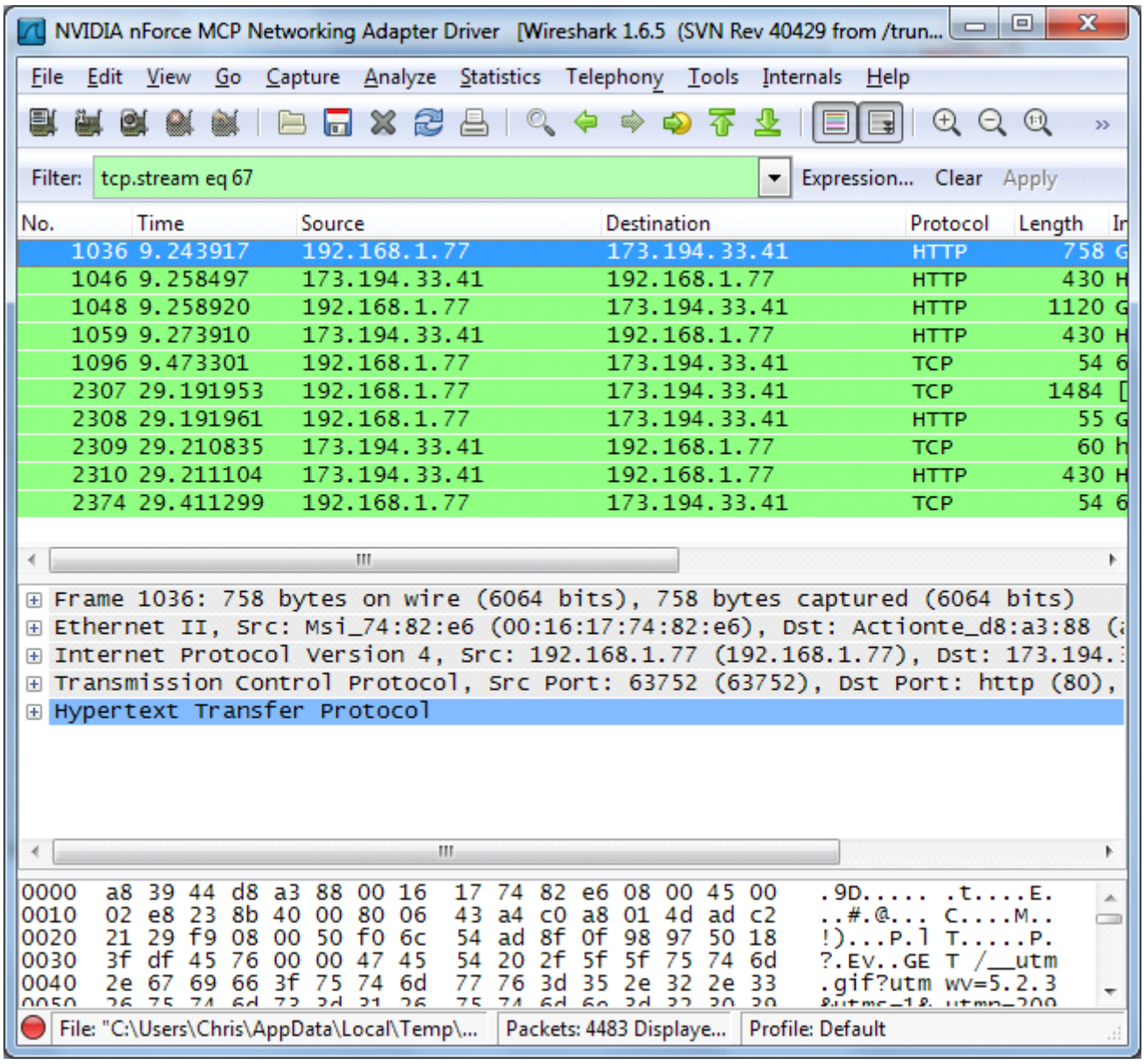
Another interesting thing you can do is right-click a packet and select Follow TCP Stream



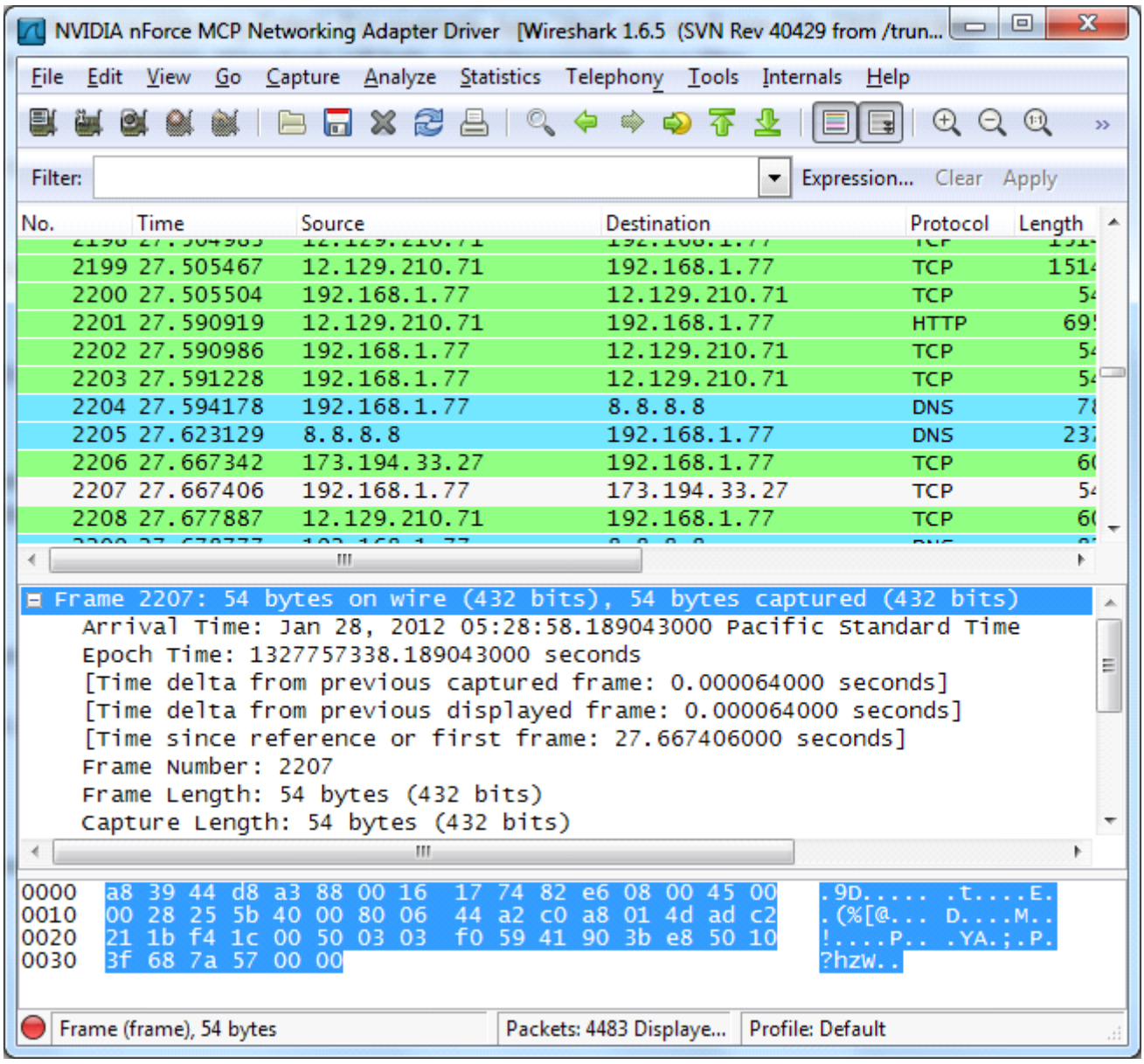
You'll see the full conversation between the client and the server.



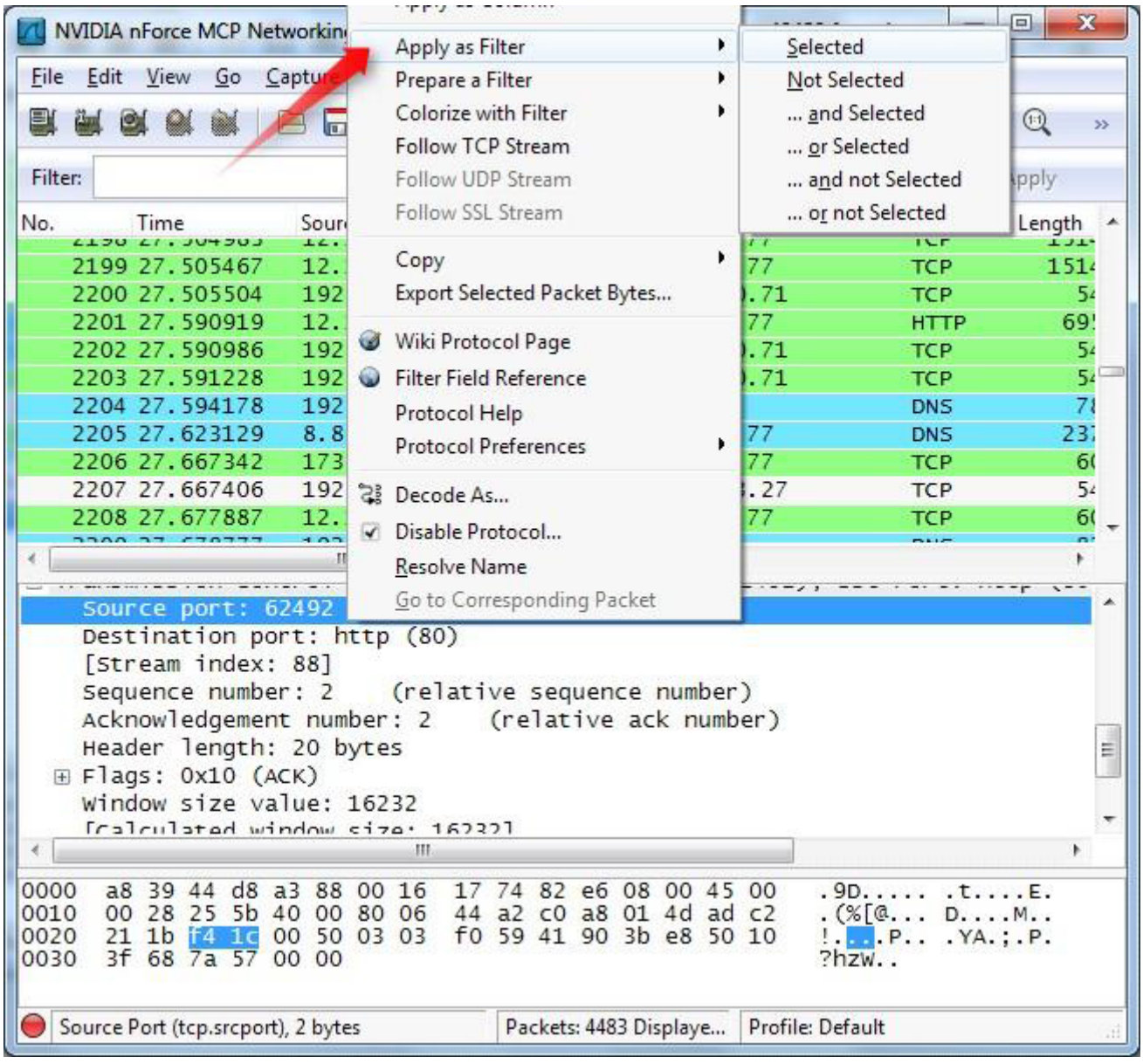
Close the window and you'll find a filter has been applied automatically — Wireshark is showing you the packets that make up the conversation.



Inspecting Packets
Click a packet to select it and you can dig down to view its details.



You can also create filters from here — just right-click one of the details and use the Apply as Filter submenu to create a filter based on it.



Wireshark is an extremely powerful tool, and this tutorial is just scratching the surface of what you can do with it. Professionals use it to debug network protocol implementations, examine security problems and inspect network protocol internals.

6. Conclusion:

In this experiment we analyze various packet sniffing tools that monitor network traffic transmitted between legitimate users or in the network. The packet sniffer is network monitoring tool. It is opted for network monitoring, traffic analysis, troubleshooting, Packet grapping, message, protocol analysis, penetration testing and many other purposes.