

5.2. Open Capture Files

[Prev](#)

Chapter 5. File Input, Output, And Printing

[Next](#)

5.2. Open Capture Files

Wireshark can read in previously saved capture files. To read them, simply select the **File** → **Open** menu or toolbar item. Wireshark will then pop up the “File Open” dialog box, which is discussed in more detail in [Section 5.2.1, “The “Open Capture File” Dialog Box”](#).



You can use drag and drop to open files

On most systems you can open a file by simply dragging it in your file manager and dropping it onto Wireshark’s main window.

If you haven’t previously saved the current capture file you will be asked to do so to prevent data loss. This warning can be disabled in the preferences.

In addition to its native file format (pcapng), Wireshark can read and write capture files from a large number of other packet capture programs as well. See [Section 5.2.2, “Input File Formats”](#) for the list of capture formats Wireshark understands.

5.2.1. The “Open Capture File” Dialog Box

The “Open Capture File” dialog box allows you to search for a capture file containing previously captured packets for display in Wireshark. The following sections show some examples of the Wireshark “Open File” dialog box. The appearance of this dialog depends on the system. However, the functionality should be the same across systems.

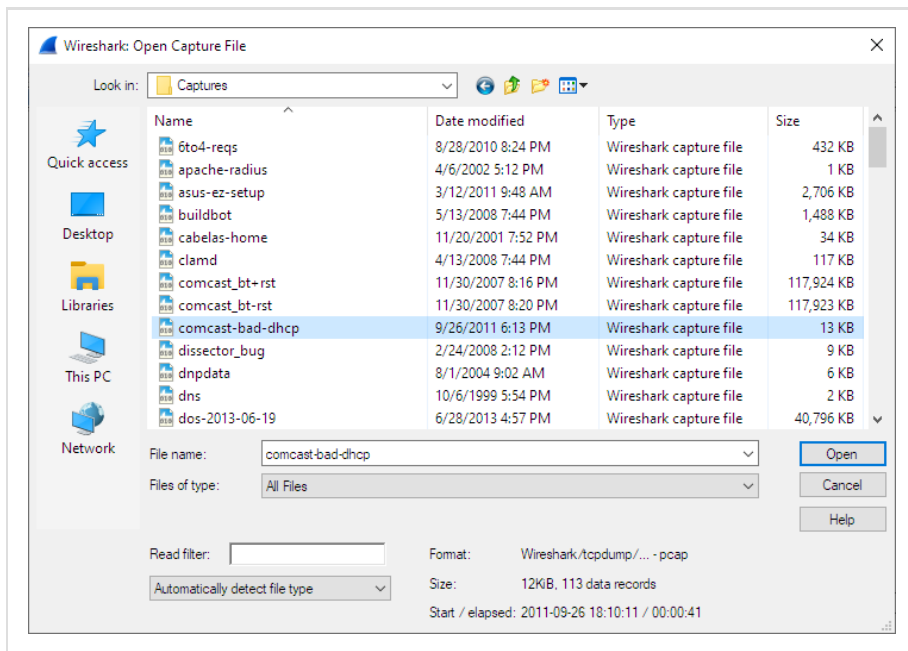
Common dialog behaviour on all systems:

- Select files and directories.
- Click the **Open** button to accept your selected file and open it.
- Click the **Cancel** button to go back to Wireshark and not load a capture file.
- The **Help** button will take you to this section of the “User’s Guide”.

Wireshark adds the following controls:

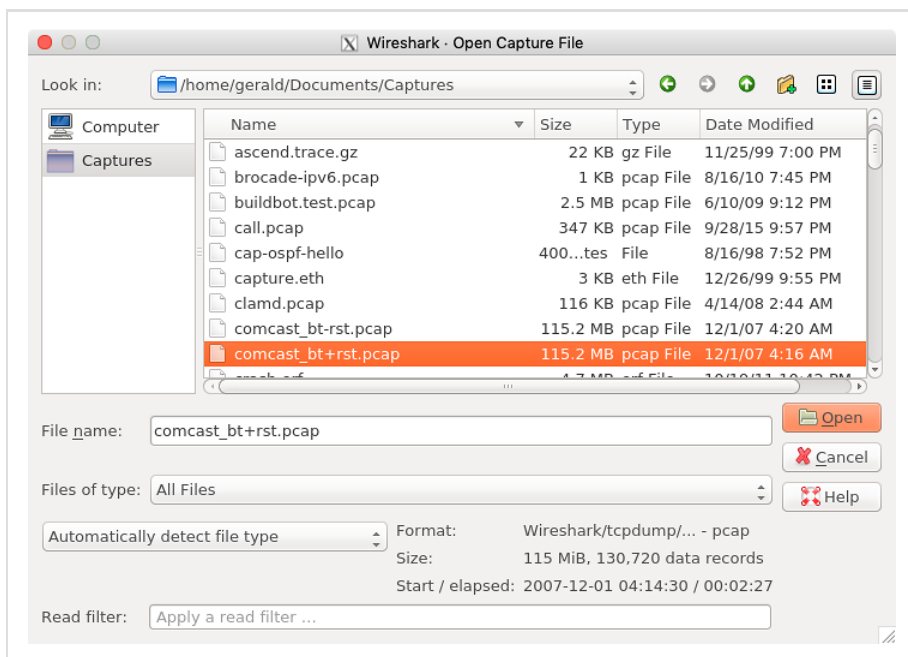
- View file preview information such as the size and the number of packets in a selected a capture file.
- Specify a read filter with the “Read filter” field. This filter will be used when opening the new file. The text field background will turn green for a valid filter string and red for an invalid one. Read filters can be used to exclude various types of traffic, which can be useful for large capture files. They use the same syntax as display filters, which are discussed in detail in [Section 6.3, “Filtering Packets While Viewing”](#).
- Optionally force Wireshark to read a file as a particular type using the “Automatically detect file type” dropdown.

Figure 5.1. “Open” on Microsoft Windows



This is the common Windows file open dialog along with some Wireshark extensions.

Figure 5.2. “Open” - Linux and UNIX



This is the common Qt file open dialog along with some Wireshark extensions.

5.2.2. Input File Formats

The native capture file formats used by Wireshark are:

- pcap. The default format used by the *libpcap* packet capture library. Used by *tcpdump*, *Snort*, *Nmap*, *Ntop*, and many other tools.
- pcapng. A flexible, extensible successor to the pcap format. Wireshark 1.8 and later save files as pcapng by default. Versions prior to 1.8 used pcap. Used by Wireshark and by *tcpdump* in newer versions of macOS.

The following file formats from other capture tools can be opened by Wireshark:

- Oracle (previously Sun) *snoop* and *atmsnoop* captures
- Finisar (previously Shomiti) *Surveyor* captures
- Microsoft *Network Monitor* captures
- Novell *LANalyzer* captures
- AIX *iptrace* captures
- Cinco Networks NetXray captures
- NETSCOUT (previously Network Associates/Network General) Windows-based Sniffer and Sniffer Pro captures
- Network General/Network Associates DOS-based Sniffer captures (compressed or uncompressed) captures
- LiveAction (previously WildPackets/Savvius) *Peek/EtherHelp/PackageGrabber captures
- RADCOM's WAN/LAN Analyzer captures
- Viavi (previously Network Instruments)i Observer captures
- Lucent/Ascend router debug output
- captures from HP-UX nettl
- Toshiba's ISDN routers dump output
- output from *i4btrace* from the ISDN4BSD project
- traces from the EyeSDN USB So
- the IPLog format output from the Cisco Secure Intrusion Detection System
- pppd logs (pppdump format)
- the output from VMS's TCPIPtrace/TCPtrace/UCX\$TRACE utilities
- the text output from the DBS Etherwatch VMS utility
- Visual Networks' Visual UpTime traffic capture
- the output from CoSine L2 debug
- the output from InfoVista (previously Accellent) 5Views LAN agents
- Endace Measurement Systems' ERF format captures
- Linux Bluez Bluetooth stack hcidump -w traces
- Catapult (now Ixia/Keysight) DCT2000 .out files
- Gammu generated text output from Nokia DCT3 phones in Netmonitor mode
- IBM Series (OS/400) Comm traces (ASCII & UNICODE)
- Juniper Netscreen snoop captures
- Symbian OS btsnoop captures
- Tamosoft CommView captures
- Tektronix K12xx 32bit .rf5 format captures
- Tektronix K12 text file format captures
- Apple PacketLogger captures
- Captures from Aethra Telecommunications' PC108 software for their test instruments
- Citrix NetScaler Trace files
- Android Logcat binary and text format logs
- Colasoft Capsa and PacketBuilder captures
- Micropross mplog files
- Unigraf DPA-400 DisplayPort AUX channel monitor traces
- 802.15.4 traces from Daintree's Sensor Network Analyzer
- MPEG-2 Transport Streams as defined in ISO/IEC 13818-1
- Log files from the *candump* utility
- Logs from the BUSMASTER tool
- Ixia IxVeriWave raw captures
- Rabbit Labs CAM Inspector files

- *systemd* journal files
- 3GPP TS 32.423 trace files

New file formats are added from time to time.

It may not be possible to read some formats dependent on the packet types captured. Ethernet captures are usually supported for most file formats but it may not be possible to read other packet types such as PPP or IEEE 802.11 from all file formats.

[Prev](#)[Up](#)[Next](#)[Chapter 5. File Input, Output, And
Printing](#)[Home](#)[5.3. Saving Captured Packets](#)