Wireshark

Wireshark is a **full-blown packet analysis workstation**. Powerful, heavy, visual.

Core Features

Live packet capture from Ethernet, Wi-Fi, loopback, Bluetooth, etc.
Deep packet inspection (Layer 2 → Layer 7)
Protocol decoding for *thousands* of protocols (HTTP, TLS, DNS, SMB, FTP, VoIP, etc.)
Automatic protocol reassembly TCP streams, HTTP conversations,VoIP calls
Advanced filtering Capture filters (BPF syntax), Display filters (Wireshark's own powerful syntax)
Color-coded packets for rapid visual analysis
Follow TCP/UDP/HTTP streams
Statistics & graphs Protocol hierarchy, Conversations, Endpoints, IO graphs
Decryption support TLS/SSL (with keys), WPA/WPA2 (with handshake + key)
Export capabilities Packets, Objects (files transferred via HTTP/FTP)
GUI-based Mouse-driven, Easy learning curve for beginners
Typical Use Cases Incident response, Malware traffic analysis, Network troubleshooting, Learning networking protocols, VoIP analysis, Forensics (pcap analysis)

Similarities

Wireshark and tcpdump are both packet capture and analysis tools.
They use the same capture engine (libpcap), support BPF filters, work on the same network layers, and can read/write .pcap files.
Their goal is the same: inspect network traffic — they only differ in interface and depth of analysis.

tcpdump

tcpdump is a **surgical knife**. Fast, raw, lethal in the right hands.

Core Features

Command-line packet capture
Uses BPF (Berkeley Packet Filter)
Extremely lightweight
Capture & save packets to .pcap
Real-time traffic inspection
Granular filtering IPs, ports, protocols, flags
Runs perfectly on servers
Can run remotely over SSH
Scriptable & automatable
No GUI dependency
Works in minimal environments
Example Capabilities Capture only SYN packets, Monitor DNS queries, Detect suspicious connections, Capture traffic during incidents without killing performance
Typical Use Cases Production servers, Cloud environments, Headless Linux systems, Automation & logging, First response during incidents